

Universidade Estadual de Campinas - UNICAMP
Faculdade de Engenharia Elétrica e de Computação

**Roteamento e Alocação de Comprimentos
de Onda com Restrições de Potência usando
Algoritmos Genéticos**

Autor: Gustavo Sousa Pavani
Orientador: Prof. Dr. Helio Waldman

Tese de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: **Telecomunicações e Telemática.**

Banca Examinadora

Prof. Dr. Helio Waldman DECOM/FEEC/UNICAMP
Prof. Dr. Michel Daoud Yacoub DECOM/FEEC/UNICAMP
Prof. Dr. Fernando José Von Zuben DCA/FEEC/UNICAMP
Prof. Dr. Amílcar Careli César DEE/EESC/USP

16 de Setembro de 2003
Campinas, SP - Brasil

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

P288r Pavani, Gustavo Sousa
Roteamento e alocação de comprimentos de onda
com restrições de potência usando algoritmos genéticos
/ Gustavo Sousa Pavani.--Campinas, SP: [s.n.], 2003.

Orientador: Helio Waldman.

Dissertação (mestrado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Comunicações óticas. 2. Algoritmos genéticos.
3. Controle ruído. I. Waldman, Helio. II. Universidade
Estadual de Campinas, Faculdade de Engenharia Elétrica
e de Computação. III. Título.

Resumo

Este trabalho visa ao estudo do problema do roteamento e alocação de comprimentos de onda (RWA) considerando restrições de potência no sinal óptico. A maioria dos trabalhos nessa área considera somente o aspecto topológico da rede, sem levar em consideração as degenerações na camada física da rede. Propõe-se um método que leva em conta a influência do ruído de emissão espontânea amplificada (ASE) na taxa de erro de bits (BER) no cálculo das restrições de potência. De forma a levar em conta a variação da imunidade de ruído através de uma cascata de amplificadores, é introduzido a noção de pré-amplificador equivalente. Somente a ASE é considerada por ser o efeito limitante para sistemas de longa distância.

A resolução se dá em duas fases: primeiro se resolve o RWA e depois uma heurística ou um algoritmo genético é usado para estabelecer o máximo número de conexões com as restrições de potência. Verifica-se a superioridade e a robustez do algoritmo genético para tal tipo de problema.

Palavras-chave Rede óptica transparente, roteamento e alocação de comprimentos de onda, degenerações na camada física, ruído ASE, algoritmos genéticos.

Abstract

This work aims at studying the routing and wavelength assignment (RWA) problem while considering constraints on the optical signal power. The majority of the previous studies in this area considered only the topological aspect of the network, not accounting for the physical impairments. We propose a method that takes into consideration the influence amplified spontaneous emission (ASE) noise on the bit error in the calculation of the power constraints. In order to take into account the noise immunity variations along the amplifier cascade, we introduce the notion of an equivalent pre-amplifier. Only the ASE noise is considered, because it is the limiting factor for long-haul systems.

Our method uses a two-phase approach: first, the RWA problem is solved and, after, a heuristic or a genetic algorithm is employed to establish the maximum number of connections under power considerations. We verified the superiority and robustness of the genetic algorithm for this kind of problem.

Keywords Transparent optical network, routing and wavelength assignment, optical layer physical impairments, ASE noise, genetic algorithms.

Sumário

Resumo	i
Abstract	iii
Sumário	v
Lista de Figuras	ix
Lista de Tabelas	xi
1 Introdução	1
1.1 Motivação	3
1.2 Escopo	4
2 Algoritmos Genéticos	5
2.1 Implementação de um Algoritmo Genético	9
2.1.1 Codificação das Soluções Candidatas	9
2.1.1.1 Codificação Binária	9
2.1.1.2 Codificação com Caracteres ou Números Reais	10
2.1.1.3 Árvores	11
2.1.2 Mecanismo de Seleção	11
2.1.2.1 Função Avaliação	12
2.1.2.2 Método de Seleção e Procedimento de Amostragem	13
2.1.2.3 Ajuste	14
2.1.3 Recombinação	15
2.1.4 Mutação	15
2.1.5 Métodos de Substituição de População	16
2.1.6 Configuração dos Parâmetros	16
2.2 Um Algoritmo Genético Simples	17
2.2.1 Representação	19
2.2.2 População Inicial	20
2.2.3 Função Avaliação	20
2.2.4 Seleção	21
2.2.5 Operadores Genéticos	22

2.2.6	Substituição de População	24
2.2.7	Parâmetros	24
2.2.8	Resultados Experimentais	25
2.3	Aplicações	25
3	A Rede Óptica	29
3.1	Histórico	29
3.2	Arquitetura da Rede	30
3.3	Degenerações do Sinal Óptico na Camada Física	31
3.3.1	Efeitos Lineares	32
3.3.1.1	Emissão Espontânea Amplificada (ASE)	32
3.3.1.2	Dispersão Cromática	33
3.3.1.3	Dispersão de Modo de Polarização (PMD)	33
3.3.2	Efeitos Não-Lineares	33
3.3.2.1	Espalhamento Estimulado de Brillouin (SBS)	34
3.3.2.2	Espalhamento Estimulado de Raman (SRS)	34
3.3.2.3	Auto Modulação de Fase (SPM)	34
3.3.2.4	Modulação Cruzada de Fase (XPM)	35
3.3.2.5	Mistura de Quatro Ondas (FWM)	35
4	Os Componentes da Rede	37
4.1	O Amplificador a Fibra Dopada a Érbio (EDFA)	37
4.1.1	Ganho	39
4.1.2	O Pré-Amplificador	41
4.1.2.1	Ruído de Batimento Sinal-ASE	44
4.1.2.2	Ruído de Batimento ASE-ASE	45
4.1.2.3	Potência Mínima	47
4.1.3	Cascata de Amplificadores: O Pré-Amplificador Equivalente	48
4.1.3.1	Caso Ideal	50
4.2	Os Demais Componentes da Rede Óptica	50
4.2.1	Terminadores Ópticos de Linha (OLT)	51
4.2.1.1	Multiplexadores	51
4.2.1.2	Demultiplexadores	51
4.2.2	Multiplexadores Ópticos de Entrada e Derivação (OADM)	51
4.2.3	<i>Cross-Connects</i> Ópticos (OXC)	51
4.2.4	Acopladores	52
4.2.5	A Fibra Óptica	52
5	Problema do Roteamento e Alocação de Comprimento de Onda (RWA)	53
5.1	RWA: Caso Estático	54
5.1.1	Subproblema do Roteamento	55
5.1.1.1	Roteamento Fixo	55
5.1.1.2	Roteamento Fixo-alternado	56

5.1.1.3	Roteamento Adaptativo	56
5.1.2	Subproblema da Alocação de Comprimento de Onda	56
5.1.2.1	Formulação matemática	57
5.2	RWA com Restrições de Potência (RWA-P)	59
5.2.1	Formulação Matemática	60
5.2.1.1	Entradas do problema	60
5.2.1.2	Variáveis	61
5.2.1.3	Restrições	64
5.2.1.4	Função Objetivo	67
5.2.2	Solução em Duas Fases	67
5.2.2.1	Mecanismo de Estabelecimento de Conexão	68
5.2.2.2	Algoritmos Genéticos	68
5.2.2.3	Heurística Menor-Ganho-Primeiro (<i>Smallest-Gain-First</i> – SGF)	69
6	Simulação e Resultados	71
6.1	Parâmetros	71
6.2	Arquitetura do Nó e da Rede	72
6.3	Os Modelos de Redes Usados	73
6.3.1	Rede Exemplo de Pequenas Dimensões	74
6.3.2	Rede Italiana de Alta Velocidade	77
7	Conclusão	81
7.1	Sensibilidade e Ruído ASE	81
7.2	Algoritmo Genético	82
7.3	Solução em Duas Fases e Uso do Caminho Mais Curto	83
7.4	Amplificadores Ópticos	84
7.5	Trabalho futuro	84
A	Algoritmos de Estabelecimento de Conexões	87
A.1	Listagens	89
B	Topologia Física da Rede Italiana	95
C	Algoritmo de Yen	101
C.1	Exemplo	103
D	Simulated Binary Crossover	107
E	Índice Remissivo de Autores	109
	Referências Bibliográficas	111

Lista de Figuras

2.1	Execução de um algoritmo genético.	7
2.2	Gráfico da função $f(x) = x \sin(10\pi x) + 1, 0$	18
4.1	O amplificador de fibra dopada a érbio.	39
4.2	Curvas de saturação para um amplificador óptico de $G_{max} = 20$ dB e $P^{sat} = 13,7$ mW.	40
4.3	Pré-amplificador e receptor.	41
4.4	Densidade espectral de potência ASE (campo óptico).	42
4.5	Densidade espectral de $x(t)$ e $y(t)$	43
4.6	Densidade espectral do ruído de batimento sinal-ASE.	45
4.7	Densidade espectral do ruído de batimento ASE-ASE.	46
4.8	Cascata de amplificadores.	48
5.1	Um enlace com 2 EDFAs e 3 segmentos de fibra. Este enlace tem 2 dispositivos e 5 componentes.	63
6.1	Arquitetura do WRN.	72
6.2	Rede exemplo.	74
6.3	Convergência do algoritmo genético para a rede exemplo com sensibilidade calculada e modelo de amplificador melhorado.	77
6.4	Rede italiana de alta velocidade.	78
6.5	Convergência do algoritmo genético para a Rede Italiana com sensibilidade calculada.	80
C.1	Exemplo do algoritmo de Yen para $v = 1$	103
C.2	Exemplo do algoritmo de Yen para $v = 2$	103
C.3	Exemplo do algoritmo de Yen para $v = 6$	104

Lista de Tabelas

2.1	Resultado ao longo de 150 gerações.	25
6.1	Parâmetros e seus respectivos valores usados no RWA-P.	71
6.2	Métodos e operadores genéticos testados.	73
6.3	Parâmetros, métodos e operadores genéticos usados nas simulações.	74
6.4	Conexões estabelecidas na rede exemplo com sensibilidade fixa.	76
6.5	Conexões estabelecidas na rede exemplo com sensibilidade calculada.	76
6.6	Conexões estabelecidas na rede exemplo com sensibilidade calculada e modelo de amplificador melhorado.	76
6.7	Matriz de tráfego para a rede italiana.	79
6.8	Conexões estabelecidas na Rede Italiana com sensibilidade fixa	79
6.9	Conexões estabelecidas na Rede Italiana com sensibilidade calculada.	79
6.10	Conexões estabelecidas na Rede Italiana com sensibilidade calculada e modelo de amplificador melhorado.	80
7.1	Convergência do algoritmo genético para a rede exemplo com sensibilidade calculada, modelo de amplificador melhorado, $W = 8$ e $K = 1$	83
B.1	Legenda da Rede Italiana.	95
C.1	As k-rotas mais curtas do exemplo do algoritmo de Yen.	105

Capítulo 1

Introdução

As redes ópticas transparentes com multiplexação por divisão de comprimento de onda (*Wavelength-Division Multiplexing* – WDM) são uma das maiores candidatas para as futuras redes de comunicação de grande distância. Nessas redes, as funções de roteamento são executadas diretamente no domínio óptico, sem a necessidade de conversão óptica-eletrônica-óptica em nós intermediários entre a origem e o destino.

Uma conexão em uma rede óptica é estabelecida através de um caminho óptico. Um caminho óptico entre duas estações é um sinal óptico que vai de um nó origem a um nó destino utilizando um comprimento de onda nos enlaces que ligam esses nós. Se não existem conversores de comprimento de onda nos nós intermediários, então um caminho óptico deve ter um mesmo comprimento de onda em todos os enlaces da origem até o destino.

Um problema muito importante nas redes ópticas é o problema do roteamento e alocação de comprimentos de onda (*Routing and Wavelength Assignment* – RWA) [52]. O objetivo do RWA é selecionar a melhor combinação de rotas e comprimentos de onda para cada conexão de forma a maximizar o número de conexões estabelecidas, de maneira tal que nenhuma conexão que compartilhe o mesmo enlace use o mesmo comprimento de onda.

Os primeiros trabalhos com o RWA [38, 33] só consideravam o aspecto lógico da rede, desconsiderando as degenerações que o sinal óptico sofre na camada física. Numa rede transparente, por não haver regeneração do sinal óptico, o ruído e a distorção do sinal vão se

acumulando ao longo do caminho óptico. Assim, em uma rede de longa distância, algumas rotas não são passíveis de serem estabelecidas, pois não possuem uma performance de transmissão aceitável. Por esse motivo, é necessário considerar essas degenerações no problema do RWA.

Aparentemente, a primeira vez em que o impacto das degenerações na camada física foram consideradas no roteamento da rede óptica foi feita por Sabella et al. [42]. Outro estudo importante [37] propôs um algoritmo de controle de admissão de chamadas baseado na taxa de erro de bits (*Bit Error Rate* – BER) integrado com o RWA; se a BER de uma conexão for maior que um certo limiar, então ela é rejeitada. Nesse artigo, a taxa de erro de bits leva em conta duas degenerações do sinal óptico: o ruído do amplificador óptico (*Amplified Spontaneous Emission* – ASE) e a interferência entre conexões causada pelo *switch*.

Esta Dissertação propõe uma extensão aos trabalhos realizados por Ali et al. em [6, 5], os quais consideram restrições de potência integradas no RWA. À semelhança de [37], um algoritmo de cálculo da BER também é proposto, o qual leva em conta o ruído ASE gerado pelos amplificadores. Neste caso, a BER determina a mínima potência que um sinal pode ter ao longo da rede. É importante notar que somente a ASE foi considerada, pois ela pode ser o fator limitante em sistemas de longa distância e é fácil de ser modelada analiticamente.

As restrições de potência podem ser definidas da seguinte forma. Um caminho óptico entre duas estações, o qual atravessa vários enlaces de fibra e encontra diversos dispositivos ópticos, deve manter níveis aceitáveis de potência ao longo de toda a sua rota. Além disso, o nível de potência de um sinal individual deve ser mantido de tal maneira que a potência agregada (isto é, a soma dos níveis de potência de todos os sinais) não pode exceder um determinado valor. O que torna esse modelo complicado é o ganho do amplificador óptico, que é dependente da potência agregada que passa por ele.

Esta Dissertação é organizada da seguinte maneira.

Capítulo 1 é esta introdução, a qual fornece um panorama deste trabalho.

Capítulo 2 introduz os algoritmos genéticos, que serão usados como heurística para a solução do problema proposto.

Capítulo 3 apresenta a rede óptica e sua arquitetura e também as degenerações de camada física mais importantes.

Capítulo 4 ilustra os componentes da rede óptica e detalha o amplificador óptico, juntamente com um modelo para estimar a BER devido a ASE.

Capítulo 5 apresenta o problema do RWA tanto no seu aspecto lógico como o modelo que considera as restrições de potência.

Capítulo 6 apresenta as redes usadas como exemplo, seus parâmetros e os resultados obtidos.

Capítulo 7 apresenta as conclusões e a proposta de trabalho futuro.

1.1 Motivação

Entre as motivações desta Dissertação, podemos citar:

Atenuação da potência do sinal óptico Vários dispositivos ópticos passivos, além da fibra óptica, atenuam o sinal: divisores, multiplexadores, elementos de permutação, etc. Duas restrições estão relacionados à potência do sinal: potência mínima e potência máxima. A restrição de potência mínima (também chamada de nível de sensibilidade) é usada para garantir que o sinal óptico possa ser detectado pelos vários dispositivos ópticos. A restrição de potência máxima é necessária para minimizar as degenerações não-lineares do sinal, que ocorrem na camada física.

Ruído Como a rede óptica transparente necessita do uso de amplificadores ópticos para compensar a atenuação do sinal pelo sistema, é necessário também considerar o ruído

na forma de emissões espontâneas (ASE), gerado por esses amplificadores. A ASE vai se acumulando à medida que vai passando pelos amplificadores ópticos, deteriorando a relação sinal-ruído, aumentando a probabilidade de erro de bit na conexão. Além disso, a ASE acumulada pode causar saturação do EDFA, diminuindo seu ganho.

Saturação do Amplificador Óptico O ganho do amplificador óptico é dependente da potência total que entra por ele, fazendo com que o ganho seja uma grandeza não-determinística dependente do tráfego [37]. Quando saturado, seu ganho cai abruptamente e isso deve ser levado em conta pelo modelo.

1.2 Escopo

Esta Dissertação se restringe ao caso estático do RWA, embora a proposta de cálculo das restrições de potência deste trabalho pode ser estendida ao caso dinâmico.

Este trabalho não leva em conta as degenerações na camada física do tipo não-linear, pois tais degenerações criam restrições que consideram rotas em conjunto, conduzindo a complexas interdependências, tais como a ordem com que se atravessa fibras específicas [45]. Além disso, o tratameno completo dessas restrições não-lineares requer um conhecimento total da infraestrutura física da rede, incluindo valores de dispersão para cada enlace de fibra, área do núcleo da fibra e sua composição, além de detalhes de subsistemas como a tecnologia de controle de dispersão. Essas informações combinadas com o estado de tráfego da rede podem determinar o nível de degeneração causado pelas não-linearidades. Claramente, isso é muito mais complicado do que o modelo proposto, que só considera a ASE como degeneração da rede.

Também este trabalho não considera outras degenerações do tipo linear como a dispersão de modo de polarização (*Polarization-Mode Dispersion* – PMD) (que é considerada em [45]) e a dispersão cromática.

Capítulo 2

Algoritmos Genéticos

Os algoritmos genéticos representam métodos de busca e otimização que foram inspirados no mecanismo de seleção natural, onde os indivíduos mais aptos têm maiores chances de sobreviver. Foram introduzidos em 1975 por John Holland da Universidade de Michigan com o objetivo de formalizar matematicamente e explicar rigorosamente processos de adaptação em sistemas naturais e desenvolver sistemas artificiais (simulados em computador) que retenham os mecanismos originais encontrados em sistemas naturais. O tema central de sua pesquisa em algoritmos genéticos foi a robustez, isto é, o balanço entre a eficiência e eficácia necessárias para a sobrevivência em diferentes ambientes [19].

Por serem uma analogia da seleção natural de Darwin, os algoritmos genéticos adotaram um vocabulário derivado da genética. Assim, um conjunto de soluções candidatas é denominado de população; cada solução candidata é denominada indivíduo ou cromossomo que por sua vez é formada por genes; cada iteração do algoritmo genético é chamado de geração; a combinação de dois (ou mais) indivíduos para se criar novos indivíduos é chamada de recombinação ou *crossover*; e a modificação aleatória de um indivíduo é chamada de mutação.

Um algoritmo genético para um determinado problema sempre deve ter os seguintes componentes [30]:

1. Uma representação genética para soluções candidatas ou potenciais (processo de codi-

- ficação);
2. Uma maneira de criar uma população inicial de soluções candidatas ou potenciais;
 3. Uma função avaliação que faz o papel da pressão do ambiente, classificando as soluções em termos de sua adaptação ao ambiente (ou seja, sua capacidade de resolver o problema);
 4. Operadores genéticos para alterar a composição dos indivíduos em uma população (em geral, crossover e mutação);
 5. Valores para os diversos parâmetros usados pelo algoritmo genético (tamanho da população, probabilidades de aplicação dos operadores genéticos, etc.).

A otimização de um problema através de algoritmos genéticos se processa da seguinte forma. Inicialmente, uma população de soluções candidatas é criada. Cada indivíduo é então avaliado pela função objetivo. Se o critério de otimização não foi atingido (definido, por exemplo, como um número pré-determinado de gerações), então a criação de uma nova população começa. Alguns indivíduos são selecionados de acordo com sua avaliação para produzir novos indivíduos através do crossover. Essa seleção se dá de forma a privilegiar a participação dos indivíduos mais adaptados. Após o crossover, os novos indivíduos sofrem mutação com uma certa probabilidade e, então, são avaliados pela função objetivo. Esses novos indivíduos são re-inseridos na população original, substituindo outros indivíduos através de algum critério de re-inserção (substituindo, por exemplo, os indivíduos de menor nível de avaliação). Esse ciclo é repetido até que o critério de otimização seja alcançado, conforme pode ser visto na Figura 2.1.

Os algoritmos genéticos são muito diferentes dos métodos tradicionais de otimização [14]. Primeiramente, os algoritmos genéticos trabalham com a codificação das variáveis em vez das próprias variáveis do problema. Dessa forma, uma codificação adequada das variáveis é necessária para o sucesso do algoritmo genético. Além disso, os algoritmos genéticos trabalham

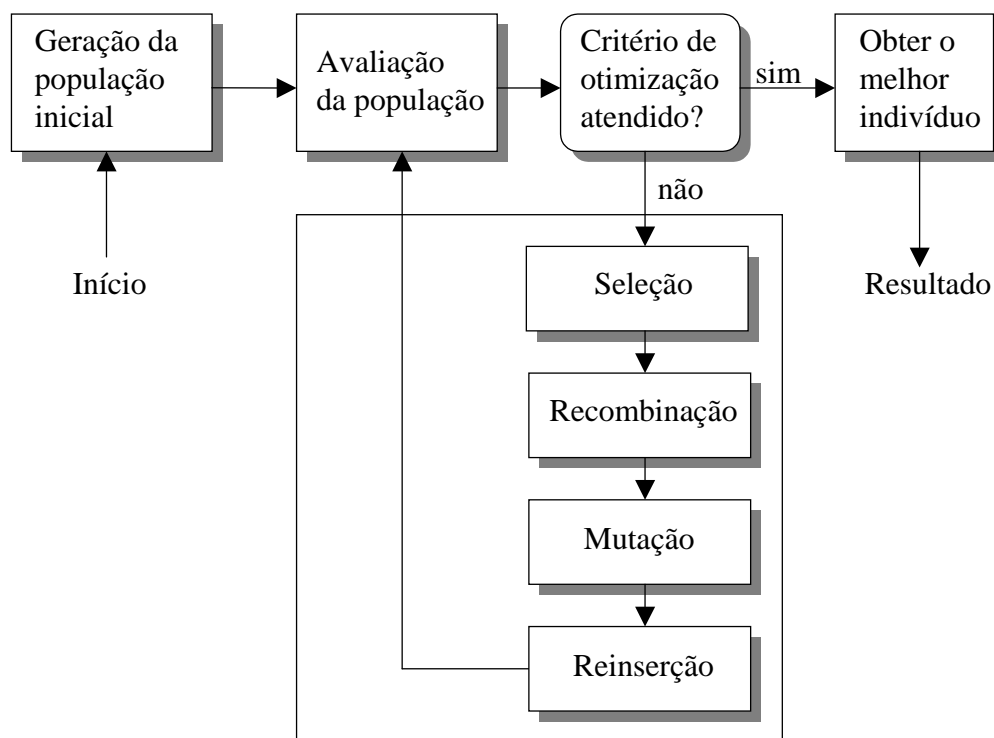


Figura 2.1: Execuç o de um algoritmo gen tico.

com uma populaç o de soluç es ao inv s de uma  nica soluç o. Outro fator importante a destacar   que os algoritmos gen ticos n o necessitam de nenhuma informa  o auxiliar al m do valor das funç o avalia  o, embora tais informa  es possam ser usadas para acelerar a converg ncia do algoritmo.

Outra grande diferen a   que os algoritmos gen ticos usam regras probabil sticas para guiar a busca, o que possibilita escapar de pontos de  timo locais, proporcionando uma robustez em uma vasta gama de problemas de otimiza  o. Ao contr rio dos m todos cl ssicos, os algoritmos gen ticos conseguem obter um equil brio entre dois objetivos aparentemente conflitantes: o aproveitamento das melhores soluç es, sem causar uma converg ncia prematura, e a explora  o do espa o de busca, sem tornar a busca totalmente cega e aleat ria. Finalmente, os algoritmos gen ticos podem ser f cil e convenientemente executados em m quinas paralelas.

Bons resultados podem ser obtidos com o uso de algoritmos genéticos, mesmo quando o problema a ser resolvido é NP-difícil ou NP-completo ou possui um espaço de busca descontínuo, não-linear, não-diferenciável, discreto, multimodo ou com presença de ruído. Na verdade, é nesses tipos de problema que os algoritmos genéticos têm mostrado um desempenho superior às técnicas convencionais, como o método do gradiente ou busca aleatória. (Para um compêndio de técnicas convencionais e não-convencionais de resolução de problemas, veja [31]).

Entretanto, é importante ressaltar que devido ao fato dos algoritmos genéticos serem uma ferramenta de busca e otimização de propósito geral, eles somente devem ser usados quando soluções clássicas ou dedicadas não existem, não se aplicam ou falham quando aplicadas. De maneira geral, os algoritmos genéticos não devem ser usados quando:

1. O espaço de busca não é muito grande, o que possibilita uma busca exaustiva da solução ótima;
2. A solução ótima deve ser necessariamente encontrada, pois não há garantias que o algoritmo genético ache a solução ótima;
3. O espaço de busca é suave ou unimodal, pois um algoritmo baseado no método do gradiente será muito mais eficiente que um algoritmo genético para explorar a suavidade do espaço de busca;
4. Se o espaço de busca é conhecido, possibilitando o uso de heurísticas específicas ao domínio da aplicação;
5. Se uma solução deve ser obtida após um certo intervalo de tempo ou há restrições de tempo real, pois embora os algoritmos genéticos possam obter uma solução sub-ótima em um curto intervalo de tempo, não há garantias de que essa solução será obtida após um certo número fixo de iterações.

Existem outros algoritmos inspirados na teoria evolutiva de Darwin, que assim como os algoritmos genéticos envolvem reprodução, variação aleatória, competição e seleção de indivíduos na população. Esses algoritmos são: Estratégia Evolutiva, Programação Evolutiva e Programação Genética. Todas essas diferentes estratégias de simular a evolução foram reunidas sob o nome de Computação Evolutiva a partir de 1991. Para um estudo mais detalhado da Computação Evolutiva, [7, 8] são excelentes referências.

2.1 Implementação de um Algoritmo Genético

A performance dos algoritmos genéticos depende de vários fatores, como a codificação das soluções candidatas, o mecanismo de seleção, os operadores e a configuração dos parâmetros. Os operadores se subdividem entre operador de recombinação e operador de mutação. Cada um desses fatores será visto separadamente a seguir.

2.1.1 Codificação das Soluções Candidatas

Como para todo método de busca ou de aprendizado, a maneira com que a solução candidata é codificada é uma fator central no sucesso de um algoritmo genético. A maioria dos algoritmos genéticos usa soluções candidatas codificadas em um arranjo de bits com tamanho fixo e ordem dos bits bem definida. Entretanto, recentemente, vários outros tipos de codificação foram usados, como a codificação usando números reais, caracteres e até mesmo outros tipos de arranjos, como árvores.

2.1.1.1 Codificação Binária

A codificação binária, isto é, uma arranjo de bits é a forma mais comum de codificação por vários motivos. O primeiro é histórico: no seu trabalho original, Holland e seus alunos concentraram-se nesse tipo de codificação e o uso de algoritmos genéticos tendem a seguir

esse trabalho. A maior parte da teoria de como os algoritmos genéticos funcionam assume que a codificação é binária com tamanho fixo e com a ordem dos bits bem definida. Grande parte dessa teoria pode ser estendida para se aplicar a codificações não-binárias, mas essas extensões não são imediatas e requerem conceituação adicional. Além disso, boas heurísticas para a configuração dos parâmetros (por exemplo, taxa de mutação e de recombinação) foram desenvolvidas no contexto da codificação binária.

Entretanto, a codificação binária não é natural nem apropriada para uma série de problemas. Assim, outras codificações foram desenvolvidas para suprir essa deficiência, como será apresentado a seguir.

2.1.1.2 Codificação com Caracteres ou Números Reais

Para muitas aplicações o mais natural é usar um alfabeto de muitos caracteres ou números reais para se formar os cromossomos. Por exemplo, para um problema com 100 variáveis com limites entre $[-500, 500]$, onde se deseja uma precisão de seis dígitos após a vírgula, o comprimento de um cromossomo codificado binariamente seria de 3000. Isso gera um espaço de busca de cerca de 10^{1000} , o que acarretaria em uma baixíssima eficiência do algoritmo genético.

Além disso, quando se usa a codificação com números reais, o algoritmo genético fica mais próximo do problema, tornando os operadores mais específicos ao problema, aumentando-se, assim, a eficiência do algoritmo genético. Por exemplo, essa representação tem como propriedade que dois pontos próximos no espaço de representação estão também próximos no espaço do problema. Isso não é necessariamente válido na codificação binária, onde a distância na representação é comumente definida pelo número de diferentes posições de bits em que duas representações diferem.

Por exemplo, no Capítulo 5 de [30], mostra-se que para a solução de um mesmo problema do tipo linear-quadrático a codificação usando números reais é bem superior à codificação

binária.

2.1.1.3 Árvores

Em alguns casos, esquemas de codificação usando árvores podem ter muitas vantagens, como o esquema de John Koza para representar programas de computador [26]. Esse esquema permite que o espaço de busca não seja limitado (a princípio, uma árvore de qualquer tamanho pode ser obtida via recombinação e mutação).

2.1.2 Mecanismo de Seleção

Após decidir sobre a codificação das soluções candidatas, o segundo passo é escolher como o algoritmo genético fará a seleção, ou seja, como se dará a escolha das soluções candidatas que irão ser usadas na criação de novas soluções e quantas novas soluções serão criadas.

O propósito da seleção é fazer com que os indivíduos mais aptos na população tenham algum tipo de prioridade na escolha para reprodução, aumentando a probabilidade de transmitir seu código genético às próximas gerações. A seleção deve ser balanceada com os operadores de recombinação e mutação. Uma seleção muito forte faz com que indivíduos sub-ótimos dominem a população, reduzindo a diversidade necessária para progressos futuros. Por outro lado, uma seleção muito fraca resulta numa evolução muito lenta.

Por conveniência, se decompõe o processo de seleção em três passos:

1. Mapear a função objetivo para a função avaliação (*fitness*);
2. Criar uma distribuição de probabilidades proporcional à avaliação;
3. Selecionar amostras de acordo com essa distribuição.

Para maiores referências ao mecanismo de seleção, veja [7, 19].

2.1.2.1 Função Avaliação

A função objetivo é definida como:

$$f : A_x \rightarrow \mathbb{R}, \quad (2.1)$$

onde A_x é o espaço das variáveis do problema (espaço de busca). A função objetivo tipicamente mede algum custo a ser minimizado ou alguma recompensa a ser maximizada. A definição da função objetivo depende da aplicação, mas há algumas linhas gerais que devem ser seguidas na escolha da função objetivo:

- A função objetivo deve refletir as características mais relevantes a serem otimizadas. Os algoritmos genéticos são notoriamente oportunistas e há muitos casos de algoritmos otimizando uma função objetivo que não representava a característica desejada.
- A função objetivo deve exibir alguma regularidade no espaço representado pela codificação das soluções candidatas.
- A função objetivo deve prover informação suficiente para guiar a busca do algoritmo genético. Por exemplo, funções do tipo “agulha no palheiro” (funções que dão quase o mesmo valor para cada solução candidata exceto para o ótimo) devem ser evitadas.

A função avaliação (*fitness function*) mapeia o valores da função objetivo para um intervalo não-negativo. É a função avaliação que é usada em última análise pelo algoritmo genético e não a função objetivo. Ela pode ser definida como:

$$\Phi : A_x \rightarrow \mathbb{R}_+ \quad (2.2)$$

A função avaliação é usada para o ajuste quando se utiliza um método de seleção proporcional à avaliação do indivíduo. Neste caso, a função avaliação é usada para se mapear a função objetivo em um intervalo não-negativo.

2.1.2.2 Método de Seleção e Procedimento de Amostragem

Dentre os métodos de seleção mais comuns, pode-se citar:

Roulette Wheel Método de seleção originalmente proposto por Holland, o qual usa uma seleção proporcional à função avaliação. O número esperado de vezes que um indivíduo será selecionado para reprodução é proporcional à sua avaliação dividida pela avaliação média da população. Para cada indivíduo é dada uma fatia de uma roleta (*roulette wheel*), proporcional à sua avaliação. Essa roleta é girada N vezes, onde N é o número de indivíduos da população. A cada giro, o indivíduo marcado na roleta é selecionado para a reprodução. Este método estocástico resulta no número esperado de filhos para cada indivíduo, mas de forma estatística. Quando se usa uma população com poucos indivíduos, o número alocado de filhos para cada indivíduo pode ficar longe de seu valor esperado.

Stochastic Universal Sampling Método de seleção proposto em [9] que melhora o método *Roulette Wheel*, minimizando a diferença entre o número alocado de filhos para cada indivíduo e o seu valor esperado.

Rank Selection Método de seleção proposto para evitar uma convergência prematura do algoritmo genético. Os indivíduos de uma população são ordenados segundo sua avaliação e o número esperado de vezes que um indivíduo será selecionado para reprodução depende da sua posição em relação aos demais indivíduos e não da sua avaliação. Após essa ordenação, o procedimento de amostragem usado pode ser o *Stochastic Universal Sampling*.

Tournament Selection Método de seleção proposto para ser mais computacionalmente eficiente que o *Rank Selection*, pois não necessita ordenar toda a população de acordo com a avaliação de cada indivíduo. Dois (ou mais) indivíduos são selecionados ale-

atoriamente da população e o “melhor” indivíduo nesse conjunto é selecionado para reprodução.

2.1.2.3 Ajuste

Os mecanismos de seleção que fazem uma seleção proporcional à avaliação do indivíduo, como o *Stochastic Universal Sampling* e o *Roulette Wheel*, precisam de um mecanismo de regulação da competição dos indivíduos durante a execução do algoritmo genético, chamado de ajuste (*scaling*). Neste caso, a função avaliação é uma composição entre a função objetivo e a função de ajuste g :

$$\Phi(a_i(t)) = g(f(a_i(t))), \quad (2.3)$$

onde $a_i(t) \in A_x$ e $g(f(a_i(t))) \in \mathbb{R}_+$.

No início da execução do algoritmo genético há uma tendência de alguns super-indivíduos (aqueles que têm uma avaliação muito superior aos demais) dominarem o processo de seleção. Neste caso, o ajuste deve reduzir a avaliação desses indivíduos para evitar uma convergência prematura do algoritmo genético. No final da execução do algoritmo genético, a população já quase convergiu, de forma que a diferença de avaliação entre os indivíduos da população é muito pequena, diminuindo assim a velocidade de convergência. Neste caso, o ajuste deve aumentar a diferença entre as avaliações dos diferentes indivíduos que compõem essa população, para continuar a recompensar os melhores indivíduos com maiores probabilidades de seleção para reprodução.

Além disso, esse ajuste também é necessário quando a meta do algoritmo genético é minimizar a função objetivo, visto que maiores valores da função avaliação correspondem a menores valores da função objetivo. Portanto, problemas de minimização devem ser transformados no seu problema de maximização equivalente, dado que a probabilidade de se selecionar um determinado indivíduo é proporcional à sua avaliação. Neste caso, uma possível função

avaliação seria:

$$\Phi(a_i(t)) = \frac{1}{1 + g(f(a_i(t)))} \quad (2.4)$$

2.1.3 Recombinação

Os indivíduos selecionados pelo mecanismo de seleção são copiados para o *mating pool*. Agora vem a fase da recombinação (*crossover*), onde os indivíduos (pais) presentes no *mating pool* são combinados de alguma forma para gerar novos indivíduos (filhos). A idéia por trás da recombinação é simples: dados dois (ou mais) indivíduos que tem uma avaliação boa, mas por diferentes razões, o ideal seria que se combinassem as melhores propriedades desses indivíduos em um único.

Em linhas gerais, dois ou mais indivíduos (pais) são selecionados no *mating pool*. Esses indivíduos são combinados com uma probabilidade igual a $p_c \in [0, 1]$ (parâmetro que indica a probabilidade de ocorrer a recombinação) para gerar um ou mais novos indivíduos. Esses novos indivíduos vão formar a população dos filhos. Com probabilidade igual a $(1 - p_c)$, os pais são copiados diretamente para a população dos filhos.

O operador de recombinação é dependente da codificação dos indivíduos.

2.1.4 Mutação

Após a fase de recombinação, cada indivíduo da população dos filhos pode sofrer mutação. A mutação introduz pequenas variações aleatórias nos genes de um indivíduo com probabilidade $p_m \in [0, 1]$.

Se os indivíduos usam uma representação binária, a mutação pode ser alcançada, por exemplo, mudando-se um bit aleatoriamente. Nas outras codificações, a mutação é um pouco mais complexa e depende dos limites do problema.

2.1.5 Métodos de Substituição de População

Após as fases de seleção, de recombinação e de mutação, os indivíduos presentes na população dos filhos devem ser inseridos na população, substituindo, de alguma forma, seus pais.

Há basicamente dois métodos em que essa substituição pode ocorrer:

1. A população dos filhos substitui completamente a população dos pais, ou seja, os filhos não competem com os pais. Esse método é conhecido como *nonoverlapping* ou geracional.
2. A população dos filhos compete com a população dos pais pela sobrevivência. Esse método é conhecido como *overlapping*.

No modelo *overlapping*, a quantidade de intersecção entre pais e filhos é chamada de *generation gap*. Esse parâmetro controla qual a fração da população será substituída a cada geração. Se o *generation gap* é de 100%, então toda a população é substituída.

Há um tipo especial no modelo *overlapping* chamado de algoritmo genético de estado estacionário (*steady state*). Nesse modelo, somente um ou dois filhos são produzidos a cada geração os quais substituem um mesmo número de pais na população.

No método *overlapping*, a escolha de quais indivíduos serão substituídos pode ser aleatória ou determinística.

Além disso, tanto no modelo *overlapping* como no modelo geracional pode ser interessante adotar uma estratégia de substituição elitista: sempre se mantém o(s) melhor(es) indivíduo(s) para a próxima geração. Isso é muito importante se o algoritmo genético é usado para otimizar uma função e o objetivo é achar o ótimo global dessa função.

2.1.6 Configuração dos Parâmetros

A última decisão para se implementar um algoritmo genético é como atribuir os valores para seus vários parâmetros, tais como o tamanho da população, a probabilidade de recombinação,

a probabilidade de mutação, etc. Esses parâmetros tipicamente interagem um com outro de maneira não-linear, de forma que não se pode otimizar um independentemente de outro.

De maneira geral, a maioria das pessoas usa os parâmetros que funcionaram bem em experiências anteriores. Na literatura, podem-se encontrar diversos valores para esses parâmetros. Por exemplo, De Jong em [24] indicou que a população deve ter de 50 a 100 indivíduos, a probabilidade de recombinação deve ser de 0,6 e a probabilidade de mutação por bit deve ser de 0,001.

Contudo, não existem princípios gerais que podem ser formulados a priori sobre a configuração dos parâmetros de um algoritmo genético, em vista da variedade dos tipos de problema, codificações e critérios de performance que são possíveis nas diferentes aplicações. Mais ainda, o tamanho da população, as probabilidades de recombinação e mutação devem mudar no curso da simulação, caso se busque um desempenho ótimo.

Dado esse problema, existe ainda muita pesquisa voltada para estratégias em que os parâmetros do algoritmo genético se adaptam durante sua execução. Para um estudo mais detalhado desse tópico, recomendamos uma leitura atenta da Parte 4 de [8].

2.2 Um Algoritmo Genético Simples

Nesta Seção, se mostrará passo a passo um algoritmo genético para a otimização de uma função de uma única variável. A função a ser otimizada (mostrada na Figura 2.2) é definida como:

$$f(x) = x \sin(10\pi x) + 1, 0 \quad (2.5)$$

O problema é achar x dentro do intervalo $[-1, 2]$ que maximiza a função $f(x)$, isto é, deve-se achar um x_0 tal que

$$f(x_0) \geq f(x) \quad , \forall x \in [-1, 2] \quad (2.6)$$

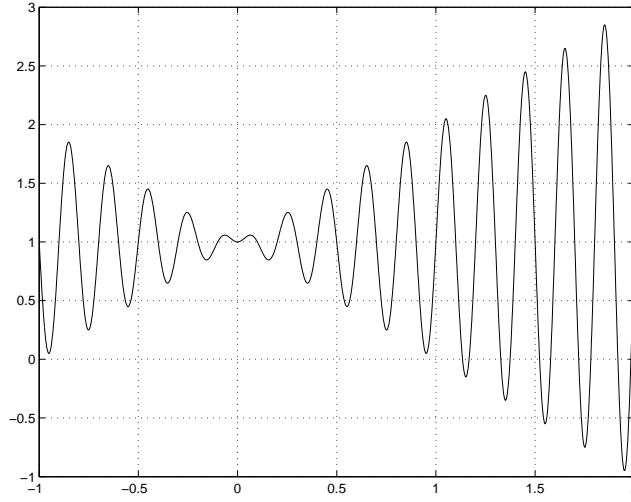


Figura 2.2: Gráfico da função $f(x) = x \sin(10\pi x) + 1,0$

É relativamente simples analisar a função $f(x)$. Os zeros da primeira derivada $f'(x)$ devem ser determinados:

$$f'(x) = \sin(10\pi x) + 10\pi x \cos(10\pi x) = 0 \quad (2.7)$$

essa expressão é equivalente a:

$$\tan(10\pi x) = -10\pi x \quad (2.8)$$

É fácil verificar que a equação acima tem um número infinito de soluções,

$$\begin{aligned} x_i &= \frac{2i-1}{20} + \epsilon_i, \text{ para } i = 1, 2, \dots \\ x_0 &= 0 \\ x_i &= \frac{2i+1}{20} - \epsilon_i, \text{ para } i = -1, -2, \dots, \end{aligned}$$

onde os termos ϵ_i representam sequências decrescentes de números reais (para $i = 1, 2, \dots$ e para $i = -1, -2, \dots$) aproximando-se de zero.

É importante notar que a função $f(x)$ alcança seu máximo local para x_i se i é um inteiro par, ou seu mínimo local para x_i se i é um inteiro ímpar.

Como o domínio do problema é $x \in [-1, 2]$, a função alcança seu máximo para $x_{19} = \frac{37}{20} + \epsilon_{19} = 1,85 + \epsilon_{19}$ onde $f(x_{19})$ é um pouco maior que $f(1,85) = 1,85 \sin(18\pi + \frac{\pi}{2}) + 1,0 = 2,85$.

Admitindo que queiramos construir um algoritmo genético para resolver o problema acima, isto é, maximizar a função $f(x)$, discutiremos os principais componentes desse algoritmo genético.

2.2.1 Representação

Iremos usar um vetor binário como cromossomo para representar os valores da variável x . O comprimento do vetor depende da precisão necessária. Para este caso, usaremos uma precisão de 6 casas após a vírgula.

O domínio da variável x tem comprimento 3; o requisito da precisão implica que o intervalo $[-1, 2]$ deve ser dividido em, no mínimo, 3×10^6 intervalos. Isso significa que 22 bits são necessários como vetor binário (cromossomo):

$$2097152 = 2^{21} < 3 \times 10^6 \leq 2^{22} = 4194304$$

O mapeamento de um arranjo binário $\langle b_{21}b_{20} \dots b_0 \rangle$ em um número real x no intervalo $[-1, 2]$ é simples e pode ser completada em 2 passos:

- converter o arranjo binário $\langle b_{21}b_{20} \dots b_0 \rangle$ da base 2 para a base 10:

$$(\langle b_{21}b_{20} \dots b_0 \rangle)_2 = \left(\sum_{i=0}^{21} b_i 2^i \right)_{10} = x'$$

- achar o correspondente número real x :

$$x = -1 + x' \frac{3}{2^{22} - 1},$$

onde -1 é o limite esquerdo do domínio e 3 é o tamanho do domínio.

Por exemplo, o cromossomo (1000101110110101000111) representa o número 0.637197, pois

$$x' = (1000101110110101000111)_2 = 2288967 \quad (2.9)$$

e

$$x = -1 + 2288967 \frac{3}{4194303} = 0,637197 \quad (2.10)$$

É fácil ver que os cromossomos (0000000000000000000000) e (11111111111111111111) representam os limites do domínio, -1 e 2 , respectivamente.

2.2.2 População Inicial

O processo de inicialização é muito simples: criamos uma população de cromossomos, onde cada cromossomo é um vetor binário de 22 bits. Cada um desses 22 bits são iniciados aleatoriamente.

2.2.3 Função Avaliação

A função avaliação *eval* para o vetor binário \mathbf{v} é equivalente à função $f(x)$:

$$eval(\mathbf{v}) = f(x) + 1, \quad (2.11)$$

onde o cromossomo \mathbf{v} representa o valor de x . Note que a função avaliação não pode resultar em um valor negativo.

Como comentado anteriormente, a função avaliação faz o papel do ambiente, classificando

as potenciais soluções em termos de sua avaliação. Por exemplo, os três cromossomos:

$$\mathbf{v}_1 = (1000101110110101000111)$$

$$\mathbf{v}_2 = (0000001110000000010000)$$

$$\mathbf{v}_3 = (1110000000111111000101),$$

correspondem aos valores $x_1 = 0,637197$, $x_2 = -0,958973$ e $x_3 = 1,67888$, respectivamente.

Consequentemente, a função avaliação vai classificá-las como:

$$eval(\mathbf{v}_1) = f(x_1) + 1 = 2,586345$$

$$eval(\mathbf{v}_2) = f(x_2) + 1 = 1,078878$$

$$eval(\mathbf{v}_3) = f(x_3) + 1 = 3,250650$$

Assim, o cromossomo \mathbf{v}_3 é o melhor dos três cromossomos, pois tem o maior valor da função objetivo.

2.2.4 Seleção

Agora que há uma população inicial e já se obteve os valores de sua função objetivo, se procede a seleção. Para este exemplo, se usará a seleção *Roulette Wheel*. Para esse mecanismo, se constrói uma roleta da seguinte maneira:

- Calcula-se o valor da função avaliação $eval(\mathbf{v}_i)$ de cada cromossomo \mathbf{v}_i , onde $i = 1, \dots, pop$ e pop é o tamanho da população.
- Calcula-se o valor total F da função avaliação $F = \sum_{i=1}^{pop} eval(\mathbf{v}_i)$
- Calcula-se a probabilidade de seleção p_i para cada cromossomo \mathbf{v}_i : $p_i = eval(\mathbf{v}_i)/F$.
- Calcula-se a probabilidade acumulada q_i para cada cromossomo \mathbf{v}_i : $q_i = \sum_{j=1}^i p_j$

A seleção é feita girando a roleta pop vezes e a cada vez se seleciona um cromossomo para *omating pool* da seguinte forma:

- Gerar um número aleatório r no intervalo $[0, 1]$;
- Se $r < q_1$, então seleciona o primeiro cromossomo. Caso contrário, seleciona-se o cromossomo \mathbf{v}_i ($2 \leq i \leq pop$) tal que $q_{i-1} < r \leq q_i$.

Supondo-se que os três vetores \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 são a população, então:

$$eval(\mathbf{v}_1) = f(x_1) + 1 = 2,586345$$

$$eval(\mathbf{v}_2) = f(x_2) + 1 = 1,078878$$

$$eval(\mathbf{v}_3) = f(x_3) + 1 = 3,250650$$

O valor total da função avaliação da população é:

$$F = \sum_{i=1}^3 eval(\mathbf{v}_i) = 6,915873$$

Dando uma probabilidade de seleção de $p_1 = eval(\mathbf{v}_1)/F = 0,373972$, $p_2 = eval(\mathbf{v}_2)/F = 0,156000$ e $p_3 = eval(\mathbf{v}_3)/F = 0,470028$.

A probabilidade acumulada de cada cromossomo é $q_1 = 0,373972$, $q_2 = 0,529972$ e $q_3 = 1,000000$.

Girando-se a roleta 3 vezes, obteve-se três números aleatórios 0,590102, 0,491215 e 0,870932. Logo, os cromossomos que vão formar o *mating pool* serão \mathbf{v}_3 , \mathbf{v}_2 e \mathbf{v}_3 .

2.2.5 Operadores Genéticos

Após a seleção, os cromossomos no *mating pool* vão sofrer a recombinação para gerar a população dos filhos. Para se fazer a recombinação dos indivíduos do *mating pool*, se procede da seguinte maneira. Para cada cromossomo no *mating pool*:

- Gerar um número aleatório r no intervalo $[0, 1]$;
- Se $r < \text{probabilidade de recombinação } (p_c)$, selecione esse cromossomo para recombinação.

Agora, pares de cromossomos selecionados são escolhidos aleatoriamente para sofrerem a recombinação. Um número inteiro no intervalo $[1, m - 1]$ (onde m é o número de bits do cromossomo) para a recombinação de um ponto. Por exemplo, supondo que os pares de cromossomos \mathbf{v}_2 e \mathbf{v}_3 foram escolhidos. Assumindo que o ponto escolhido aleatoriamente foi 5, o ponto de recombinação será depois do quinto gene:

$$\mathbf{v}_2 = (00000 | 01110000000010000)$$

$$\mathbf{v}_3 = (11100 | 00000111111000101)$$

Resultando em dois filhos que são:

$$\mathbf{v}'_2 = (00000 | 00000111111000101)$$

$$\mathbf{v}'_3 = (11100 | 01110000000010000)$$

Esses dois filhos são avaliados como:

$$\text{eval}(\mathbf{v}'_2) = f(-0,998113) + 1 = 1,940865$$

$$\text{eval}(\mathbf{v}'_3) = f(1,666028) + 1 = 3,459245$$

Note que o segundo filho tem um maior valor da função objetivo que ambos os pais.

Após a recombinação, o próximo operador a ser aplicado é a mutação. Para cada cromossomo na nova população gerada pelo *crossover* e para cada bit dentro do cromossomo:

- Gerar um número aleatório r no intervalo $[0, 1]$;

- Se $r < \text{probabilidade de mutação } (p_m)$, fazer a mutação desse bit.

Entretanto, se $p_m \ll 1$, recomenda-se o uso do algoritmo presente no Capítulo 24 de [8].

A mutação pode alterar um ou mais genes (posições do cromossomo) com um probabilidade igual à probabilidade de mutação. Supondo que o quinto gene do cromossomo \mathbf{v}'_3 foi selecionado para mutação. Logo o cromossomo \mathbf{v}'_3 depois da mutação torna-se:

$$\mathbf{v}''_3 = (1110\mathbf{1}01110000000010000)$$

Esse cromossomo representa o valor de $x'_3 = 1,759778$ e $eval(x'_3) = 0,677401$. Isso significa que essa particular mutação resultou num significativo decréscimo do valor do cromossomo \mathbf{v}'_3 . Por outro lado, se o nono gene for selecionado para mutação no cromossomo \mathbf{v}'_3 , então:

$$\mathbf{v}'''_3 = (1110001\mathbf{1}00000000010000)$$

O valor correspondente de $x'''_3 = 1,660168$ e $eval(x'''_3) = 3,576178$, resultando em uma melhora sobre o valor original de $eval(x'_3) = 3,459245$.

2.2.6 Substituição de População

Após a fase de mutação, a população dos filhos substitui completamente a população dos pais, ou seja, este algoritmo genético é geracional e usa elitismo.

2.2.7 Parâmetros

Para este exemplo, usaram-se os seguintes parâmetros [30]:

- Tamanho da população (pop) = 50
- Probabilidade de recombinação (p_c) = 0.25

- probabilidade de mutação (p_m) = 0.01

2.2.8 Resultados Experimentais

Neste exemplo, o algoritmo genético teve 150 iterações (gerações). Na Tabela 2.1 é fornecida a geração em que houve uma melhora na função objetivo, junto como o valor dessa função.

O melhor cromossomo após 150 gerações foi:

$$\mathbf{v}_{max} = (1111001101000100000101),$$

o qual corresponde ao valor $x_{max} = 1.850773$.

Como esperado, $x_{max} = 1.85 + \epsilon$, e $f(x_{max})$ é um pouco maior que 2.85.

Geração	Função objetivo
1	1,441942
6	2,250003
8	2,250283
9	2,250284
10	2,20363
12	2,328077
39	2,344251
40	2,345087
51	2,738930
99	2,849246
137	2,850217
150	2,850227

Tabela 2.1: Resultado ao longo de 150 gerações.

2.3 Aplicações

Os algoritmos genéticos já foram usados em um grande número de problemas científicos e da engenharia. Alguns exemplos [32]:

Otimização Os algoritmos genéticos são usados em uma grande variedade de tarefas, incluindo otimização numérica e otimização de problemas combinatórios, às vezes, com vantagens sobre outras técnicas de otimização.. Entre esses vários empregos, há projeto de filtros, projeto de controladores, projeto VLSI, sistemas de reconhecimento de voz, planejamento de produção e problemas de agendamento, projeto de redes de comunicação, roteamento, etc. Para uma descrição mais detalhada, veja [27, 7].

Programação autônoma Os algoritmos genéticos são usados para fazer a evolução de programas de computador que executam tarefas específicas e para compor estruturas computacionais como redes de ordenação e autômatos celulares.

Aprendizado de máquina Os algoritmos genéticos são usados em várias aplicações de aprendizado de máquina, incluindo tarefas de classificação e previsão, como previsão do tempo e estruturas de proteína. Também são usados para melhorar sistemas de aprendizado de máquina, evoluindo os pesos de redes neurais, regras de sistemas classificadores ou de sistemas de produção simbólica e sensores de robôs.

Economia Os algoritmos genéticos são usados para modelar processos de inovação, o desenvolvimento de estratégias de leilão e a emergência de mercados.

Sistemas imunes Os algoritmos genéticos são usados para modelar vários aspectos do sistema imune, incluindo mutação somática durante a vida do indivíduo, a descoberta de famílias de genes múltiplos durante a evolução, clonagem e maturação de afinidade

Ecologia Os algoritmos genéticos são usados para modelar fenômenos ecológicos como co-evolução de parasita-hospedeiro, simbiose e fluxo de recursos.

Genética de populações Os algoritmos genéticos são usados para estudar as questões relativas à genética de populações, como “sob quais condições um gene para recombinação será viável?”

Aprendizado e evolução Os algoritmos genéticos são usados para estudar como a aprendizagem de um indivíduo e a evolução das espécies afetam um ao outro.

Sistemas sociais Os algoritmos genéticos são usados para estudar os aspectos evolucionários de sistemas sociais, como a evolução do comportamento social de colônias de insetos e a evolução da cooperação e comunicação em sistemas multi-agentes.

Essa lista não é extensa, mas ela dá uma idéia de como os algoritmos genéticos podem ser usados. Por causa do seu sucesso nessas e em outras áreas, o interesse em algoritmos genéticos está aumentando rapidamente. Os algoritmos genéticos se tornaram uma disciplina da Computação, com conferências, periódicos e sociedades científicas próprias.

Capítulo 3

A Rede Óptica

Neste Capítulo apresentar-se-á o histórico das redes ópticas, sua arquitetura e principais problemas de degeneração do sinal óptico na camada física.

3.1 Histórico

A fase de pesquisa em sistemas ópticos de comunicação começou ao redor de 1975. Dependendo do autor, há várias formas de se classificar as diferentes gerações de redes ópticas. Nesta Dissertação, usar-se-ão as gerações definidas por Agrawal em [3].

A primeira geração de sistemas ópticos de comunicação operavam em torno do comprimento de onda de $0,8 \mu\text{m}$ com taxas de 45 Mb/s , usavam lasers semicondutores de GaAs, utilizavam repetidores a cada 10 km e começaram a ser comercialmente disponíveis a partir de 1980.

A segunda geração de redes ópticas usavam lasers de InGaAsP que operavam em torno da frequência de $1,3 \mu\text{m}$, devido à menor dispersão da fibra em torno dessa frequência. No início, se usavam fibras multimodo que limitavam as taxas de transmissão a 100 Mb/s , que foram substituídas por fibras monomodo as quais permitiam taxas de até $1,7 \text{ Gb/s}$ com repetidores a cada 50 km , por volta de 1987.

A terceira geração de sistemas ópticos operam em torno da frequência de $1,55 \mu\text{m}$ e em

1990 se chegou a taxas de até 10 Gb/s com repetidores a cada 60-70 km.

A quarta geração faz uso de amplificação óptica para aumentar o espaço entre repetidores e de multiplexação por divisão de comprimentos de onda (*Wavelength-Division Multiplexing* – WDM) para aumentar as taxas de transmissão. Na maioria dos sistemas WDM, as perdas na fibra são compensadas pelo uso de amplificadores de fibra dopada a érbio (EDFA), espaçados entre 60-80 km. Tais sistemas começaram a ser desenvolvidos em 1985 e se tornaram comercialmente disponíveis em 1990.

A quinta e atual geração tenta aumentar a largura de banda disponível e também aumentar a taxa de transmissão por canal. Atualmente se trabalha com taxas de 40 Gb/s com tendência para passar a taxas de 160 Gb/s.

3.2 Arquitetura da Rede

A rede de comunicação óptica é conhecida por suas altas taxas de transmissão. Entretanto, a rede óptica é capaz de prover muito mais funcionalidades do que somente uma conexão ponto-a-ponto rápida. O advento das redes de quarta geração, as quais fazem uso de EDFAs, possibilitou que o roteamento e processamento de pacotes, que eram antes restritos completamente ao domínio eletrônico, fossem feitos no domínio óptico. Redes ópticas baseadas nesse paradigma são chamadas de redes roteadas por comprimento de onda, que no restante dessa Dissertação serão referidas somente como redes ópticas ou sistemas de comunicação ópticos.

A rede óptica fornece um caminho óptico para um usuário, por exemplo, um terminal SONET ou um roteador IP. Caminhos ópticos são conexões ópticas que vão de um nó origem a um nó destino utilizando um comprimento de onda nos enlaces que ligam esses nós. Em alguns casos, pode haver conversão de um comprimento de onda para outro em um mesmo caminho óptico.

Uma importante funcionalidade da rede óptica é que o caminho óptico pode ser transpa-

rente para os dados que trafegam por ele, uma vez estabelecido. Assim, o caminho óptico pode servir uma conexão independentemente de sua taxa de bits (desde que não se ultrapasse os limites desse caminho) ou do formato do protocolo. Pode servir também uma conexão analógica.

Outro termo associado à transparência é a rede totalmente óptica, na qual os dados são levados da origem até seu destino no domínio óptico, sem nunca sofrer uma conversão óptica-eletrônica-óptica em nenhum ponto da rede. Entretanto, a rede totalmente óptica é limitada no seu escopo pelas degradações do sinal óptico na camada física, as quais serão estudadas na Seção a seguir. Se o sinal óptico necessita de regeneradores ao longo do caminho óptico, a rede deixa de ser transparente e se torna opaca, fazendo com que a rede só consiga usar uma única taxa de transmissão e um único tipo de protocolo.

Esta Dissertação assume uma rede totalmente óptica e transparente, que leva em conta as degenerações da camada física no seu gerenciamento e controle, de forma a manter um nível aceitável na taxa de erro de bits (*Bit Error Rate* – BER) e ao mesmo tempo não fazer uso de regeneradores. O escopo desta Dissertação é limitado ao estudo da degradação do sinal óptico imposto pelo ruído de Emissão Espontânea Amplificada (ASE) do EDFA, desconsiderando-se outros tipos de degeneração na camada física da rede óptica, embora essas degenerações sejam também apresentadas na Seção a seguir.

3.3 Degenerações do Sinal Óptico na Camada Física

As degenerações do sinal óptico na camada física afetam o roteamento na rede óptica e por isso devem ser consideradas [45]. Esta Dissertação se restringe a estudar o efeito do ruído ASE no roteamento da rede óptica, pois ele é o efeito limitante em sistemas de longa distância devido ao cascadeamento de EDFAs. Entretanto, por caráter informativo, uma descrição sucinta de outros efeitos que aparecem em sistemas WDM serão também apresentados.

Os efeitos que degradam o sinal óptico na camada física podem ser divididos em duas categorias: lineares e não-lineares. Os efeitos lineares são independentes da potência do sinal e não causam interferência em outros canais. Já os efeitos não-lineares são fortemente relacionados aos níveis de potência e causam interferência entre os canais WDM.

3.3.1 Efeitos Lineares

Os efeitos lineares que ocorrem em redes ópticas são basicamente três: emissão espontânea amplificada (ASE), dispersão cromática e dispersão de modo de polarização (PMD).

3.3.1.1 Emissão Espontânea Amplificada (ASE)

A emissão espontânea amplificada (*Amplified Spontaneous Emission* – ASE) é o ruído dominante gerado pelo EDFA. Essa emissão dá origem a um largo espectro de fótons que são amplificados junto com o sinal óptico.

O processo de emissão espontânea não contribui para o ganho de primeira ordem do amplificador. Embora os fótons emitidos tenham a mesma energia que os fótons incidentes do sinal óptico, eles são emitidos em direção, polarização e fase aleatórios. Esse processo é diferente do processo de emissão estimulada, onde os fótons emitidos têm a mesma direção, polarização e fase dos fótons incidentes, e que é responsável pela amplificação do sinal óptico.

A emissão espontânea prejudica o sistema óptico. O amplificador não consegue distinguir o sinal óptico das emissões espontâneas, amplificando ambas. Assim, em redes ópticas de grande distâncias, que se valem do cascadeamento de amplificadores ópticos para compensar as perdas do sistema, a ASE se torna o fator mais crítico que deve ser gerenciado. Há duas razões para isso [3]: a ASE vai se acumulando pelos amplificadores e degrada a relação sinal-ruído quando o número de amplificadores aumenta. A segunda razão é que, conforme o nível de ASE aumenta, ela contribui para saturar os amplificadores ópticos, reduzindo, dessa forma, o ganho dos amplificadores que estão localizados mais a frente no enlace. O resultado

final é que o sinal tende a diminuir e a ASE tende a aumentar em sistemas de longa distância, quando se vai aumentando o número de amplificadores ópticos, até que, em um caso limite, a taxa de erros de bit (BER) fique inaceitável.

3.3.1.2 Dispersão Cromática

Dispersão cromática é o nome dado ao fenômeno em que diferentes componentes espectrais de um pulso se propagam a diferentes velocidades. Ela acontece por dois motivos. Primeiro, o índice de refração da sílica, que é o material com que é feita a fibra óptica, é dependente da frequência. Esse componente da dispersão cromática é chamado de dispersão material. A segunda razão é que o índice efetivo de refração do modo fundamental é também dependente da frequência. Esse segundo componente é chamado de dispersão da guia de onda.

A dispersão cromática faz com que a forma dos pulsos propagando na fibra não seja preservada, provocando, em geral, um alargamento dos mesmos.

3.3.1.3 Dispersão de Modo de Polarização (PMD)

A dispersão de modo de polarização (*Polarization-Mode Dispersion* – PMD) acontece porque as fibras não são perfeitamente simétricas, de modo que dois modos ortogonalmente polarizados tenham diferentes constantes de propagação, ou seja, as fibras são levemente birefringentes. Como a energia do pulso propagando na fibra é geralmente dividido entre esses modos, essa birefringência causa um alargamento do pulso.

3.3.2 Efeitos Não-Lineares

Há duas categorias de efeitos não-lineares. A primeira categoria compreende os efeitos que surgem devido à interação das ondas de luz com os fónons (vibrações moleculares) na sílica. Devido ao espalhamento presente nesses efeitos, a energia de uma onda de luz é transmitida para uma segunda onda, chamada de onda de Stokes, que tem maior comprimento de onda.

A energia perdida é absorvida pelo fónons. A primeira onda funciona como um bombeio que causa uma amplificação da onda de Stokes. Os dois principais efeitos dessa categoria são o espalhamento estimulado de Brillouin (SBS) e espalhamento estimulado de Raman (SRS).

O segundo tipo de efeitos não-lineares ocorre devido à dependência do índice de refração à potência óptica na fibra. Os principais efeitos dessa categoria são: auto modulação de fase (SPM), modulação cruzada de fase (XPM) e mistura de quatro ondas (FWM).

Para uma completa referência sobre os efeitos não-lineares, veja [2, 3].

3.3.2.1 Espalhamento Estimulado de Brillouin (SBS)

No espalhamento estimulado de Brillouin (*Stimulated Brillouin Scattering* – SBS) os fónons acústicos estão envolvidos na interação de espalhamento, e essa interação ocorre em uma faixa de cerca de 20 MHz (no comprimento de onda $1,55 \mu\text{m}$). Neste caso, a onda que funciona como bombeio é um sinal óptico e a onda de Stokes é uma onda gerada devido ao processo de espalhamento, a qual se propaga no sentido oposto à onda que serve de bombeio. SBS pode causar grande distorção dentro de um mesmo canal (intracanal).

3.3.2.2 Espalhamento Estimulado de Raman (SRS)

No espalhamento estimulado de Raman (*Stimulated Raman Scattering* – SRS) o efeito tem maior alcance que o SBS: canais espaçados de até 15 THz podem ser afetados pelo SRS. Neste caso, a onda que funciona como bombeio é um sinal de menor comprimento de onda e a onda de Stokes é um sinal de maior comprimento de onda.

3.3.2.3 Auto Modulação de Fase (SPM)

No caso da auto modulação de fase (*Self-Phase Modulation* – SPM) os sinais transmitidos sofrem gorgieio (*chirping*). Isto acontece porque o índice de refração da fibra causa um componente que é dependente da intensidade. Esse índice de refração não-linear causa um

deslocamento de fase que é proporcional à intensidade do pulso. Como diferentes partes do pulso possuem diferentes deslocamentos de fase, surge o *chirping* dos pulsos. O *chirp* induzido pela SPM pode aumentar bastante a largura do pulso devido à dispersão cromática.

3.3.2.4 Modulação Cruzada de Fase (XPM)

Nos sistemas WDM, os efeitos não-lineares são aumentados uma vez que os sinais combinados de todos os canais podem ter uma potência muito alta. Dessa forma, o deslocamento de fase dependente da intensidade do pulso com o consequente *chirping* induzidos pelo SPM é aumentado pela intensidade dos outros canais. Esse efeito é conhecido como modulação cruzada de fase (*Cross-Phase Modulation* – XPM).

3.3.2.5 Mistura de Quatro Ondas (FWM)

Em um sistema WDM que consiste no conjunto de frequências f_1, \dots, f_n , a mistura de quatro ondas (*Four-Wave Mixing* – FWM) gera novos sinais nas frequências com $2f_i - f_j$ e $f_i + f_j - f_k$. Essa interferência entre canais é particularmente severa quando o espaçamento entre os canais é pequeno. Além disso, uma baixa dispersão cromática aumenta a interferência entre canais gerados pela FWM.

Capítulo 4

Os Componentes da Rede

A rede óptica é formada por inúmeros componentes. Dentre os mais comumente usados podemos citar:

- O amplificador óptico;
- A fibra;
- O *switch*;
- O multiplexador;
- O demultiplexador;
- O acoplador.

Para esta Dissertação é necessário um estudo mais detalhado do amplificador óptico, o qual ocupará a maior parte deste Capítulo. Os demais componentes serão sucintamente apresentados após a Seção do amplificador óptico.

4.1 O Amplificador a Fibra Dopada a Érbio (EDFA)

Os amplificadores ópticos se tornaram um componente essencial nos sistemas de transmissão e redes para compensar as perdas do sistema. Num sistema óptico de comunicação , o

sinal vai sendo atenuado pela fibra óptica à medida que ele vai se propagando. Outros componentes ópticos, como multiplexadores e acopladores também causam uma atenuação do sinal. Após certa distância, a perda na potência do sinal faz com que ele seja impossível de ser corretamente detectado. Assim, a potência do sinal deve ser restaurada antes que ela não fique reduzida demais. Antes do advento dos amplificadores ópticos, isso era feito com o uso de regeneradores, os quais convertiam o sinal para o domínio elétrico e depois retransmitiam o sinal óptico novamente pela fibra óptica. Os regeneradores possuem vários inconvenientes, como a dependência da taxa de bits e do tipo da modulação e a necessidade de se ter um regenerador para cada comprimento de onda.

O amplificador óptico mais comum atualmente é o amplificador à fibra dopada a érbio (EDFA), o qual opera na banda C (1530–1565 nm). Além disso, EDFAs na banda L (1565–1625 nm) e amplificadores Raman também são usados. EDFAs são usados em quase todos os sistemas WDM, enquanto os amplificadores Raman são usados em conjunto com os EDFAs em vários sistemas de longuíssima distância, como enlaces transoceânicos.

O EDFA consiste de uma fibra de sílica cujo núcleo é dopado com íons de érbio (Er^{3+}). A fibra recebe um laser de bombeio com o comprimento de onda de 980 ou 1480 nm. Para se juntar o laser de bombeio com o sinal óptico a ser amplificado, há um acoplador com seleção de comprimentos de onda no início da fibra dopada.

Na sua saída, outro acoplador com seleção de comprimentos de onda pode ser usado para separar o sinal amplificado dos vestígios do sinal de bombeio. Em geral, um isolador é usado na entrada e/ou saída para evitar reflexões dentro do amplificador. Um esquema possível de EDFA é mostrado na Figura 4.1.

Os amplificadores podem ser usados em três tipos diferentes de configuração:

- O pré-amplificador é usado antes de um receptor para melhorar sua sensibilidade.
- O amplificador de potência é usado após o transmissor para aumentar a potência de saída.

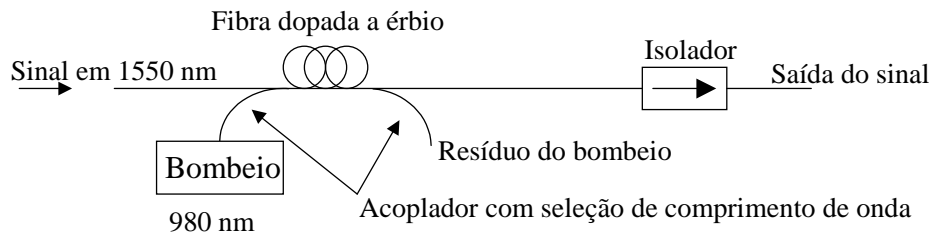


Figura 4.1: O amplificador de fibra dopada a érbio.

- O amplificador de linha é usado no enlace para compensar as perdas de atenuação de seus componentes.

Entre os motivos que garantiram o sucesso do uso dos EDFAs, pode-se citar:

1. A disponibilidade de lasers semicondutores de bombeio que são extremamente compactos e confiáveis;
2. Como o EDFA é feito de fibra, ele é independente da polarização e facilmente acoplável nos sistemas ópticos;
3. A simplicidade do dispositivo;
4. Largura de banda de 25 nm na região do comprimento de onda de 1550 nm, com relativa planura de ganho nessa região. Isso possibilita a amplificação de vários canais WDM simultaneamente.
5. O fato de não introduzir interferência entre canais na amplificação de sinais WDM.

4.1.1 Ganho

O ganho no EDFA depende da potência total dos sinais ópticos de entrada. Para altas potências de entrada, o EDFA satura e seu ganho diminui. Dessa forma, os sistemas que usam o EDFA devem levar em conta a sua saturação.

Esse comportamento pode ser aproximado pela seguinte equação [22]:

$$G = 1 + \frac{P^{sat}}{P_{in}} \ln \frac{G_{max}}{G}, \quad (4.1)$$

onde G_{max} é o ganho não-saturado ou ganho de sinal pequeno, P^{sat} é a potência interna de saturação do amplificador e P_{in} é a potência do sinal de entrada. A potência interna de saturação é uma função da potência de bombeio e de outros parâmetros do amplificador e é da ordem de 10 a 100 mW.

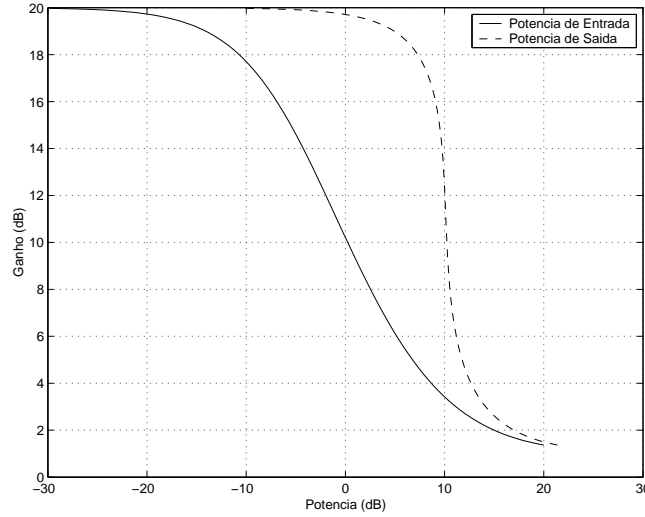


Figura 4.2: Curvas de saturação para um amplificador óptico de $G_{max} = 20$ dB e $P^{sat} = 13,7$ mW.

Para baixos valores de potência de entrada, o ganho tende a ser o ganho de sinal pequeno. Para altos valores de potência de entrada, o ganho tende a ficar igual a 1 e a potência de saída igual à potência de entrada (vide Figura 4.2).

Como a Equação 4.1 é transcendental, ela deve ser resolvida numericamente. Uma forma eficiente de resolvê-la é através do método de Newton-Raphson, bastante usado para aproximar as raízes de uma equação do tipo $f(x) = 0$. Esse método gera uma sequência $\{p_n\}$,

$n = 1, 2, \dots, \infty$, definida por:

$$p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)}, \quad (4.2)$$

onde $f'(p_n)$ é a derivada de $f(p_n)$.

Rescrevendo a Equação 4.1 como:

$$f(G) = G - 1 - \frac{P_{sat}}{P_{in}} \ln \frac{G_{max}}{G} = 0, \quad (4.3)$$

temos que sua derivada é igual a:

$$f'(G) = 1 + \frac{P_{sat}}{P_{in}G} \quad (4.4)$$

Considerando $p_0 = G_{max}$, basta aplicar a Equação 4.2 usando as Equações 4.3 e 4.4 repetidamente até que $p_{n+1} - p_n < TOL$, onde TOL é a aproximação desejada.

4.1.2 O Pré-Amplificador

Será examinado agora a imunidade ao ruído do conjunto pré-amplificador/receptor eletrônico (veja Figura 4.3), com o intuito de determinar a potência mínima necessária na entrada do pré-amplificador de forma a se obter um desempenho satisfatório. Para isso, suporemos que o ganho do amplificador seja suficiente para sobrepujar todos os ruídos internos ao receptor (ruído térmico e ruído balístico), de maneira que o desempenho será limitado apenas pela ASE.

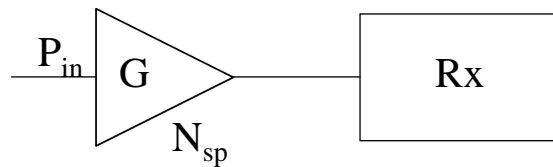


Figura 4.3: Pré-amplificador e receptor.

A potência da ASE na saída do pré-amplificador é dada por [39]:

$$P_{ASE} = N_{sp} h f_c (G - 1) B_o, \quad (4.5)$$

onde:

N_{sp} é o fator de emissão espontânea. Idealmente é igual a 1, mas na prática varia de 2 a 4.

f_c é a frequência da portadora óptica.

h é a constante de Planck.

$\therefore h f_c$ é a energia dos fótons.

G é o ganho do amplificador.

B_o é a faixa óptica, igual à faixa do filtro óptico em sistemas WDM.

O espectro da ASE se estende por uma faixa de largura B_o em torno da frequência f_c . Supondo um espectro plano, teremos a densidade espectral da ASE como representada na Figura 4.4.

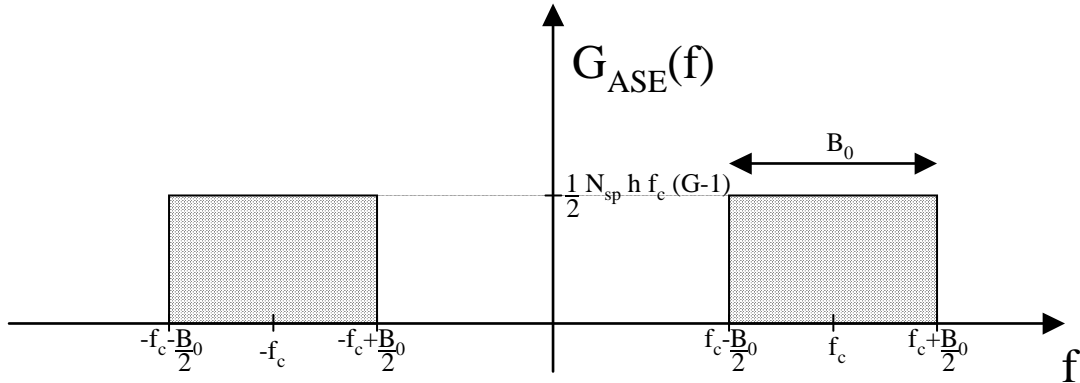


Figura 4.4: Densidade espectral de potência ASE (campo óptico).

Pode se considerar a ASE como tendo uma distribuição gaussiana por se tratar de um grande número de contribuições independentes e identicamente distribuídas. Assim sendo, o

campo da ASE pode ser assim representado:

$$n(t) = x(t) \cos(2\pi f_c t) + y(t) \sin(2\pi f_c t), \quad (4.6)$$

onde $x(t)$ e $y(t)$ são processos gaussianos independentes de banda básica. Suas densidades espectrais podem ser representadas conforme mostrado na Figura 4.5.

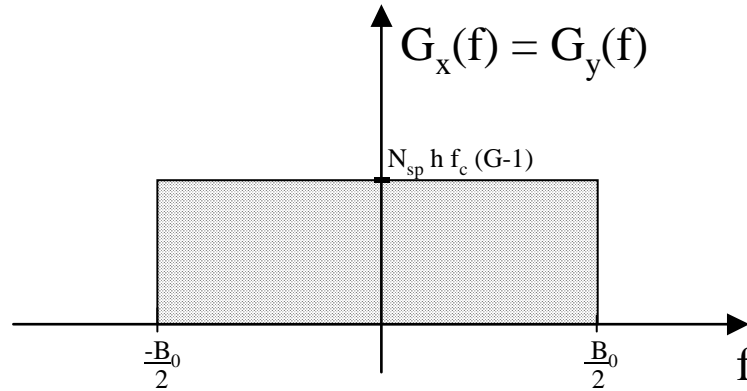


Figura 4.5: Densidade espectral de $x(t)$ e $y(t)$.

A componente $x(t)$ está em fase com o sinal e a componente $y(t)$ em quadratura. O sinal pode ser representado como abaixo:

$$s(t) = \begin{cases} A \cos(2\pi f_c t) & , \text{ bit 1} \\ 0 & , \text{ bit 0} \end{cases},$$

onde $\frac{A^2}{2} = GP_{in}$ e P_{in} é a potência da marca na entrada do pré-amplificador.

O sinal contaminado por ASE na entrada do receptor será dado por:

$$z(t) = [A + x(t)] \cos(2\pi f_c t) + y(t) \sin(2\pi f_c t) \quad (4.7)$$

O estágio frontal do receptor é constituído pelo fotodetector, que gera uma corrente

proporcional à potência de $z(t)$. Tomando-se a potência média sobre intervalos muito maiores que f_c^{-1} , mas bem menores que B_o^{-1} , teremos:

$$\begin{aligned}
 q(t) &= z^2(t) \\
 &= \frac{1}{2}[A + x(t)]^2 + \frac{1}{2}y^2(t) \\
 &= \underbrace{\frac{A^2}{2}}_{\text{sinai}} + \underbrace{Ax(t)}_{\text{sinai-ASE}} + \underbrace{\frac{x^2(t) + y^2(t)}{2}}_{\text{ASE-ASE}}
 \end{aligned}$$

O primeiro termo ($\frac{A^2}{2} = GP_{in}$) representa o sinal. O segundo termo ($Ax(t) = \sqrt{2GP_{in}}x(t)$) representa o ruído de batimento sinal-ASE. O terceiro termo representa o ruído de batimento ASE-ASE.

4.1.2.1 Ruído de Batimento Sinal-ASE

Sendo resultado da multiplicação de $x(t)$ por $A = \sqrt{2GP_{in}}$, o espectro deste ruído é simplesmente o produto de $G_x(f)$ por $2GP_{in}$.

Antes de chegar ao estágio de decisão, porém, o ruído é submetido à filtragem elétrica. Trata-se de uma filtragem passa-baixa de faixa B_e , com o intuito de eliminar todo o ruído fora da faixa do sinal. Sendo B a taxa de bits, a faixa elétrica B_e deve ser pelo menos igual a $B/2$ (critério de Nyquist), mas na prática é mais próxima de B para melhorar a abertura lateral do diagrama de olho [25].

Da Figura 4.6, temos a potência do ruído de batimento no ponto de decisão igual a:

$$\sigma_{\text{signal-ASE}}^2 = 4P_{in}N_{sp}hf_cG(G-1)B_e \quad (4.8)$$

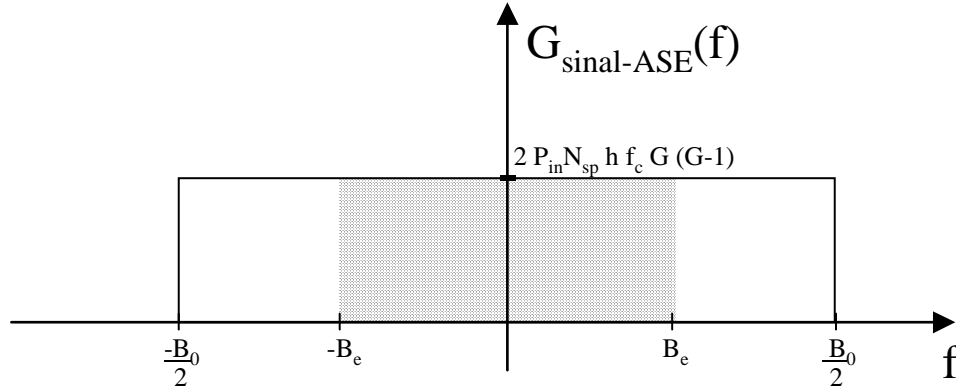


Figura 4.6: Densidade espectral do ruído de batimento sinal-ASE.

4.1.2.2 Ruído de Batimento ASE-ASE

Como $x(t)$ e $y(t)$ são processos independentes e identicamente distribuídos, o espectro do ruído ASE-ASE $\frac{x^2(t)+y^2(t)}{2}$ é igual ao espectro de x^2 :

$$\begin{aligned} G_{ASE-ASE}(f) &= G_{x^2}(f) \\ &= \mathcal{F}\{R_{x^2}(\tau)\}, \end{aligned}$$

onde $R_{x^2}(\tau) = \overline{x^2(t)x^2(t+\tau)} = \overline{x(t)x(t)x(t+\tau)x(t+\tau)}$

De acordo com a Teoria dos Processos Gaussianos, sendo x_1, x_2, x_3, x_4 processos gaussianos de média zero, temos [35]:

$$\overline{x_1 x_2 x_3 x_4} = \overline{x_1 x_2 x_3 x_4} + \overline{x_1 x_3 x_2 x_4} + \overline{x_1 x_4 x_2 x_3}$$

Portanto:

$$\begin{aligned} R_{x^2}(\tau) &= \overline{x^2(t)x^2(t+\tau)} + 2\overline{x^2(t)x^2(t+\tau)} \\ &= \overline{x^2(t)}^2 + 2R_x(\tau)^2 \end{aligned}$$

Procedendo à transformação de Fourier, temos:

$$G_{x^2}(f) = \overline{x^2(t)}^2 \delta(f) + 2G_x(f) * G_x(f)$$

Como o ruído ASE-ASE é independente do sinal, a componente DC é irrelevante, pois aparece tanto nos zeros como nas marcas. Fazendo-se a autoconvolução de $G_x(f)$, resulta então o espectro $G_{x^2}(f)$, que é mostrado na Figura 4.7.

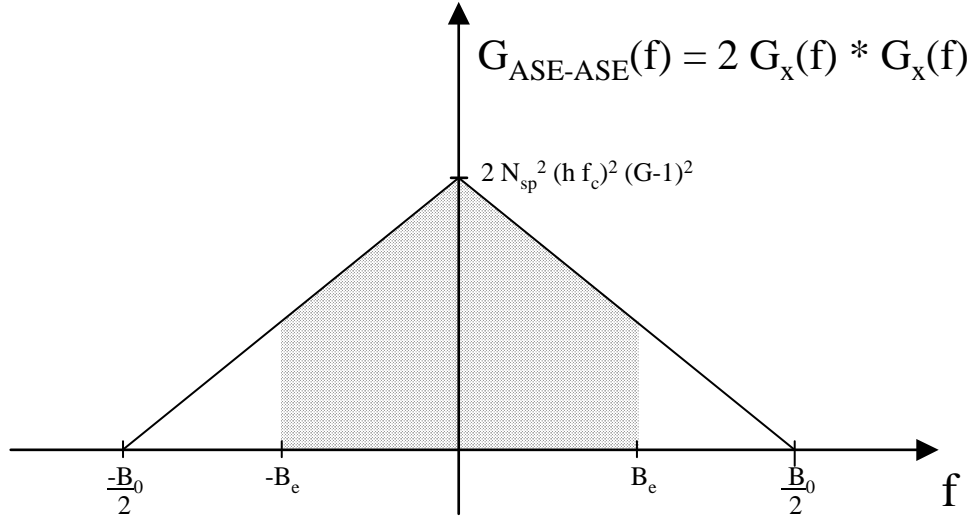


Figura 4.7: Densidade espectral do ruído de batimento ASE-ASE.

A potência do ruído de batimento ASE-ASE no ponto de decisão resulta da integração deste espectro sobre a faixa elétrica, correspondendo à área do trapézio mostrado na figura. Temos, dessa forma:

$$\begin{aligned} \sigma_{ASE-ASE}^2 &= 2N_{sp}^2(hf_c)^2(G-1)^2 B_o \left(1 - \frac{B_e}{2B_o}\right) 2B_e \\ &= 4N_{sp}^2(hf_c)^2(G-1)^2 \left(B_o - \frac{B_e}{2}\right) B_e \end{aligned}$$

No ponto de decisão, temos então os seguintes valores para o sinal e ruídos:

$$signal = \begin{cases} GP_{in} & , \text{ bit 1} \\ 0 & , \text{ bit 0} \end{cases} \quad (4.9)$$

$$\text{ruído sinal-ASE} = \sigma_{sinal-ASE} = 2\sqrt{P_{in}N_{sp}hf_cG(G-1)B_e} \quad (4.10)$$

$$\text{ruído ASE-ASE} = \sigma_{ASE-ASE} = 2N_{sp}hf_c(G-1)\sqrt{B_e\left(B_o - \frac{B_e}{2}\right)} \quad (4.11)$$

4.1.2.3 Potência Mínima

Supondo um limiar de decisão P_{sen} igual à metade do sinal de entrada P_{in} , este deverá ser no mínimo igual a $2\gamma\sigma$, onde σ é o ruído RMS (suposto gaussiano) e $Q(\gamma) = P_e$ é a probabilidade de erro tolerável [25], onde:

$$Q(t) = \int_t^\infty \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}} dx \quad (4.12)$$

O fator Q é comumente usado para especificar a performance do receptor, pois ele está relacionado a uma razão sinal-ruído (SNR) que é necessária para se alcançar determinada taxa de erro de bits (BER). Por exemplo, para uma BER de 10^{-15} temos, aproximadamente, $Q = 8$. Atualmente, se considera uma BER de 10^{-15} para sistemas WDM novos.

$$\begin{aligned} \therefore GP_{in} &\geq 2\gamma\sigma = 2\gamma\sqrt{\sigma_{sinal-ASE}^2 + \sigma_{ASE-ASE}^2} \\ \therefore G^2P_{in}^2 &\geq 4\gamma^2 \left\{ 4P_{in}N_{sp}hf_cG(G-1)B_e + 4N_{sp}^2(hf_c)^2(G-1)^2B_e \left(B_o - \frac{B_e}{2} \right) \right\} \end{aligned}$$

Fazendo $G - 1 \approx G$, pois $G \gg 1$, temos:

$$P_{in}^2 - (16\gamma^2N_{sp}hf_cB_e)P_{in} - 16\gamma^2(hf_c)^2B_e \left(B_o - \frac{B_e}{2} \right) \geq 0$$

Assim:

$$\begin{aligned}
 P_{in} &\geq \frac{1}{2} \left\{ 16\gamma^2 N_{sp} h f_c B_e + \sqrt{256\gamma^4 N_{sp}^2 (h f_c)^2 B_e^2 + 64\gamma^2 N_{sp}^2 (h f_c)^2 B_e \left(B_o - \frac{B_e}{2} \right)} \right\} \\
 \therefore P_{in} &\geq 8\gamma^2 N_{sp} h f_c B_e \left\{ 1 + \sqrt{1 + \frac{\frac{B_o}{B_e} - \frac{1}{2}}{4\gamma^2}} \right\}
 \end{aligned} \tag{4.13}$$

Como $P_{in} \geq 2P_{sen}$, temos:

$$\therefore P_{sen} = 4\gamma^2 N_{sp} h f_c B_e \left\{ 1 + \sqrt{1 + \frac{\frac{B_o}{B_e} - \frac{1}{2}}{4\gamma^2}} \right\} \tag{4.14}$$

4.1.3 Cascata de Amplificadores: O Pré-Amplificador Equivalente

Supondo agora que o receptor receba o sinal de uma linha equipada com uma cascata de amplificadores de linha, numerados de $1, 2, \dots$ a partir do receptor, como é mostrado na Figura 4.8. O pré-amplificador pode ser considerado como o amplificador número 0 da cascata. Seja G_i o ganho do amplificador i e $N_{sp\ i}$ o seu fator de emissão espontânea. O trecho de linha entre o i -ésimo e o $(i - 1)$ -ésimo amplificador tem atenuação L_i . Seja ainda P_i a potência da marca na entrada do i -ésimo amplificador.

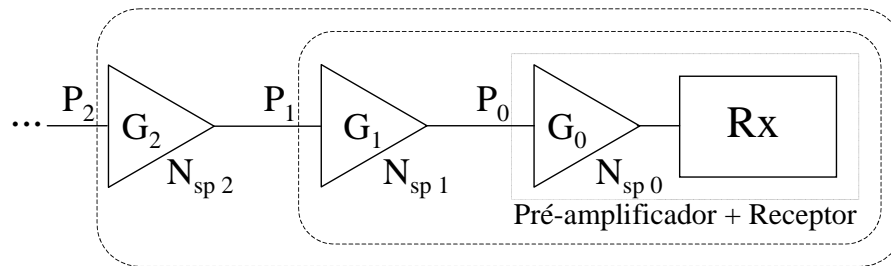


Figura 4.8: Cascata de amplificadores.

O valor mínimo de P_i pode ser obtido da Equação 4.13 representando o trecho da cascata, da entrada do amplificador i até a saída do pré-amplificador, como um pré-amplificador

equivalente de ganho G_i^{eq} e fator de emissão espontânea $N_{sp\ i}^{eq}$. Isso pode ser feito pela seguinte identificação:

$$\begin{aligned} G_i^{eq} &= G_i L_i G_{i-1} L_{i-1} \dots G_1 L_1 G_0 \\ &= \left(\prod_{j=0}^i G_j \right) \left(\prod_{j=1}^i L_j \right) \end{aligned} \quad (4.15)$$

Aplicando-se a Equação 4.15 na Equação 4.5, obtemos:

$$\begin{aligned} P_{ASE} &= (hf_c) B_o \left\{ N_{sp\ i} (G_i - 1) \left(\prod_{j=0}^i G_j \right) \left(\prod_{j=1}^i L_j \right) + \dots \right\} \\ &= (hf_c) B_o N_{sp\ i}^{eq} (G_i^{eq} - 1) \end{aligned} \quad (4.16)$$

O cálculo pode ser feito por recorrência, gerando os valores mínimos de P_1 , P_2 , P_3 , etc. Assim, para o amplificador 1, teremos:

$$G_1^{eq} = G_1 L_1 G_0 \quad (4.17)$$

$$N_{sp\ 1}^{eq} (G_1^{eq} - 1) = N_{sp\ 1} (G_1 - 1) L_1 G_0 + N_{sp\ 0} (G_0 - 1) \quad (4.18)$$

Através das Equações 4.17 e 4.18, é possível obter a fórmula final para o cálculo do fator de emissão espontânea equivalente:

$$\therefore N_{sp\ 1}^{eq} = \frac{N_{sp\ 1} (G_1 - 1) L_1 G_0 + N_{sp\ 0} (G_0 - 1)}{G_1 L_1 G_0 - 1} \quad (4.19)$$

O valor de $N_{sp\ i}^{eq}$ entra no lugar de N_{sp} na Equação 4.5 para calcular P_1 . Em dB, o

acréscimo de potência em relação a P_0 é dado por:

$$\begin{aligned} 10 \log_{10} \frac{P_1}{P_0} &= 10 \log_{10} \frac{N_{sp1}^{eq}}{N_{sp0}} \\ &= 10 \log_{10} \frac{\left(\frac{N_{sp1}}{N_{sp0}} \right) (G_1 - 1) L_1 G_0 + G_0 - 1}{G_1 L_1 G_0 - 1} \end{aligned} \quad (4.20)$$

4.1.3.1 Caso Ideal

Supondo $G_1 = G_0 = G$, $L_1 = 1/G$ e $N_{sp1} = N_{sp0} = N_{sp}$, o acréscimo será de 3 dB e $N_{sp1}^{eq} = 2N_{sp0}$.

Aplicando-se a Equação 4.19 recorrentemente para obter P_2 , P_3 , etc. . . e supondo $G_i = G \forall i$, $L_i = 1/G \forall i$ e $N_{spi} = N_{sp} \forall i$, teremos:

$$N_{spi}^{eq} = i N_{sp} \quad (4.21)$$

Resultando em:

$$P_i \geq P_0 [dBm] + 10 \log i \quad (\text{em dBm}) \quad (4.22)$$

Assim sendo, após 10 amplificadores idênticos em cascata, por exemplo, a potência mínima seria 10 dB acima do que seria exigido na entrada do pré-amplificador, devido ao acúmulo da ASE gerada pelos 11 amplificadores.

4.2 Os Demais Componentes da Rede Óptica

Esta Seção fornecerá uma descrição sucinta dos demais componentes da rede óptica. Para maiores informações sobre esses e outros componentes, uma excelente referência é [39].

4.2.1 Terminadores Ópticos de Linha (OLT)

Os Terminadores Ópticos de Linha (*Optical Line Terminals* – OLT) são usados nas extremidades de um enlace WDM ponto-a-ponto. Podem ser de dois tipos: multiplexadores e demultiplexadores.

4.2.1.1 Multiplexadores

Os multiplexadores multiplexam vários comprimentos de onda em uma única fibra.

4.2.1.2 Demultiplexadores

Os demultiplexadores demultiplexam vários comprimentos de onda, que estão em uma única fibra, em várias fibras.

4.2.2 Multiplexadores Ópticos de Entrada e Derivação (OADM)

Multiplexadores Ópticos de Entrada e Derivação (*Optical Add/Drop Multiplexers* – OADM) recebem vários sinais em múltiplos comprimentos de onda e seletivamente derivam alguns comprimentos de onda enquanto deixam passar os demais. Ele também é capaz de adicionar comprimentos de onda. Um OADM tem 2 portas de linha, onde estão presentes os canais WDM e um certo número de portas locais, em que comprimentos de onda individuais são adicionados ou derivados.

4.2.3 *Cross-Connects* Ópticos (OXC)

O *Cross-Connect* Óptico (OXC) tem a mesma funcionalidade que o OADM, mas possui um número maior de portas, que podem chegar a várias dezenas, e são capazes de rotear um comprimento de onda de uma porta de entrada para uma porta de saída. Tanto OADMs como OXCs podem ter recursos de conversão de comprimento de onda. Todavia, esta Dissertação

assume OXC sem capacidade de conversão de comprimentos de onda e chama a matriz de permutação (*Wavelength Routing Switches* – WRS) do OXC simplesmente de *switch*.

4.2.4 Acopladores

Um acoplador é usado para combinar ou dividir sinais em uma rede óptica. O acoplador tem vários usos na rede óptica, por exemplo, no acoplamento do sinal de bombeio com o sinal proveniente do enlace no EDFA.

4.2.5 A Fibra Óptica

A fibra óptica é um excelente meio de transmissão se comparada com o cobre ou o ar livre. A fibra permite uma transmissão com baixas perdas com uma faixa de pelo menos 25 THz. Em geral, as fibras têm como parâmetro de fabricação o seu coeficiente de atenuação (α), que é cerca de 0,2 dB/km para um comprimento de onda na região de 1,55 μm , mas que é bastante alto (cerca de 5 dB/km) para a luz visível.

Entretanto, a resposta da fibra, como qualquer meio dielétrico à luz se torna não linear quando submetida a campos eletromagnéticos intensos, causando a maior parte das degenerações (veja Seção 3.3) que o sinal óptico sofre na rede óptica.

Capítulo 5

Problema do Roteamento e Alocação de Comprimento de Onda (RWA)

A técnica de multiplexação por divisão em comprimentos de onda (WDM) nas redes ópticas é o meio mais usado para se tratar o crescimento da demanda nessas redes. Numa rede WDM roteada por comprimentos de onda, a comunicação é feita através de canais ópticos WDM, que são chamados de caminhos ópticos (*lightpaths*).

Um caminho óptico define uma conexão em uma rede WDM, podendo passar por diversas fibras. Se a rede não possui conversores de comprimento de onda, então um caminho óptico deve ter sempre o mesmo comprimento de onda em todos os segmentos de fibra pelos quais passa. Essa propriedade é chamada de restrição de continuidade de comprimento de onda. Esta Dissertação considera que não há conversores de comprimento de onda presentes na rede WDM. Logo esta restrição é adotada ao longo deste trabalho.

Dado um conjunto de conexões, o problema de configurar caminhos ópticos através do roteamento e alocação de um comprimento de onda para cada conexão é chamado de problema do Roteamento e Alocação de Comprimento de Onda (*Routing and Wavelength Assignment* – RWA) [52]. Tipicamente, as requisições de conexão podem ser de 3 tipos: estáticas, incrementais ou dinâmicas.

Com o tráfego estático, todo o conjunto de conexões é conhecido com antecedência e o problema se restringe a estabelecer os caminhos ópticos para essas conexões de uma maneira

global enquanto se minimiza o uso de recursos da rede como o número de comprimentos de onda ou o número de fibras na rede. Uma alternativa é tentar estabelecer o número máximo dessas conexões dado um conjunto fixo de comprimentos de onda disponível. O RWA para o tráfego estático é conhecido como problema do Estabelecimento de Caminhos Ópticos Estáticos (*Static Lightpath Establishment* – SLE).

No caso do tráfego incremental, as requisições de conexão chegam sequencialmente e um caminho óptico é estabelecido para cada conexão, com cada caminho óptico ficando ativo na rede indefinidamente.

E no caso do tráfego dinâmico, um caminho óptico é estabelecido à medida que as requisições de conexões chegam e cada caminho óptico é finalizado depois de algum tempo finito, que pode ser aleatório.

O objetivo dos casos de tráfego incremental e dinâmico é estabelecer caminhos ópticos e alocar comprimentos de onda de forma a minimizar a quantidade de bloqueio de conexões ou maximizar o número de conexões que são estabelecidas na rede a qualquer momento. Esse problema é conhecido como Estabelecimento de Caminhos Ópticos Dinâmicos (*Dynamic Lightpath Establishment* – DLE).

Esta Dissertação se restringirá ao caso estático (SLE) do RWA, o qual será apresentado com mais detalhes na próxima Seção.

5.1 RWA: Caso Estático

Nesta Seção, será apresentado o problema do RWA para o caso estático (SLE). No SLE as requisições para caminho óptico são conhecidas com antecedência e as operações de roteamento e alocação de comprimentos de onda são feitas *off-line*. O objetivo mais comum é minimizar o número de comprimentos de onda necessários para estabelecer um certo conjunto de caminhos ópticos para uma dada topologia física. Uma alternativa à minimização de

comprimentos de onda é maximizar o número de conexões que podem ser estabelecidas para um dado número de comprimentos de onda e um dado conjunto de requisições de conexão. É esse segundo objetivo que será usado nesta Dissertação.

O SLE, quando se considera a restrição de continuidade de comprimentos de onda, pode ser formulado como um problema de programação inteiro-linear (ILP)[38], que é, entretanto, NP-completo [13].

Embora o problema do roteamento e alocação de comprimentos de onda seja NP-completo, limitando sua aplicação a pequenas instâncias, ele pode ser simplificado dividindo-se esse problema em dois subproblemas distintos: o subproblema do roteamento e o subproblema da alocação de comprimento de onda.

5.1.1 Subproblema do Roteamento

O subproblema do roteamento se restringe a achar uma rota ou caminho para um par origem-destino de uma requisição de conexão. Há basicamente 3 tipos de roteamento:

5.1.1.1 Roteamento Fixo

É a maneira mais simples de se rotear, pois sempre se escolhe uma mesma rota para um dado par origem-destino. Um exemplo de tal estratégia é o roteamento pela rota mais curta. A rota mais curta para cada par origem-destino é calculada usando algoritmos de caminho mais curto, como o algoritmo de Dijkstra [4] ou de Bellman-Ford, e qualquer conexão entre os nós é estabelecida usando uma rota pré-determinada. Essa estratégia, contudo, tem uma desvantagem: no caso dinâmico ela causa altas probabilidades de bloqueio e no caso estático ela pode usar um grande número de comprimentos de onda.

5.1.1.2 Roteamento Fixo-alternado

Essa estratégia considera múltiplas rotas entre pares origem-destino. Dessa forma, deve-se manter uma lista ordenada de rotas para um par-origem destino. Por exemplo, essas rotas podem incluir a rota mais curta, a segunda rota mais curta, a terceira rota mais curta, etc. Em geral, a rota mais curta é a rota primária e as demais são consideradas como alternativas. Essas rotas podem ser calculadas, por exemplo, com o algoritmo de Yen [51], o qual é capaz de achar as k rotas mais curtas e que não contêm *loops*, embora essas rotas não são necessariamente disjuntas.

5.1.1.3 Roteamento Adaptativo

Nessa estratégia, a rota de um nó origem a um nó destino é escolhida dinamicamente, dependendo do estado da rede. O estado da rede é determinado pelo conjunto de todas as conexões que estão ativas.

5.1.2 Subproblema da Alocação de Comprimento de Onda

No subproblema da alocação de comprimento de onda deve-se alocar um comprimento de onda para cada caminho óptico de tal forma que dois caminhos ópticos não compartilham um mesmo comprimento de onda em uma mesma fibra, dado um conjunto de requisições de caminhos ópticos e suas rotas. Uma maneira possível de se resolver esse problema é formulá-lo como um problema de coloração de grafos. Outra forma é usar a formulação que emprega a programação inteira-linear (ILP) do RWA. Neste caso, a restrição de continuidade de comprimentos de onda está implicitamente definida, pois as rotas já foram pré-calculadas no sub-problema do roteamento.

A formulação que se segue considera que já se resolveu o subproblema de roteamento e se empregou a estratégia de roteamento fixa-alternada (vide 5.1.1.2) com uso do algoritmo de Yen [51].

5.1.2.1 Formulação matemática

O problema pode ser descrito numa formulação inteira-linear que é mostrada a seguir:

Entradas do problema:

- A topologia física da rede, descrita como um grafo bi-direcional $G = (V, E)$, onde V é o conjunto formado por estações e *cross-connects* e E é o conjunto de enlaces da rede;
- O conjunto de estações N da rede ($N \subset V$);
- Matriz de demanda $T_{|N| \times |N|}$, o qual especifica as requisições de conexão entre as estações. $T_{s,d}$ é o número de requisições de conexão da estação de origem s à estação de destino d ;
- O conjunto de comprimentos W disponíveis em um enlace;
- O conjunto de rotas $P_{s,d}$ para cada par origem-destino (s, d) . É o resultado obtido pelo algoritmo de Yen;
- As matrizes de constantes de pertinência rota-enlace $V_l^{s,d,r}$. Note que l é um enlace e r é uma rota. Tais matrizes derivam das rotas obtidas pelo algoritmo de Yen.

$$V_l^{s,d,r} = \begin{cases} 1 & : l \in r, r \in P_{s,d} \\ 0 & : \text{caso contrário} \end{cases}$$

Variáveis:

- Variável inteira $\sigma_{s,d}^r$ indicando o número de caminhos ópticos para o par (s, d) roteado na r -ésima rota;

- Variável binária $\Lambda_{c,l}$ indicando o uso do comprimento de onda c no enlace l .

$$\Lambda_{c,l} = \begin{cases} 1 & : \text{se alguma conexão usa o comprimento de onda } c \text{ no enlace } l \\ 0 & : \text{caso contrário} \end{cases}$$

- Variável binária $\lambda_{c,r}^{s,d}$ indicando o uso de um comprimento de onda em uma dada rota.

Note que esta variável é independente do enlace l .

$$\lambda_{c,r}^{s,d} = \begin{cases} 1 & : \text{se o comprimento de onda } c \text{ é usado pela } r\text{-ésima rota entre o par} \\ & \text{origem-destino } (s, d), \text{ onde } 1 \leq r \leq |P_{s,d}| \\ 0 & : \text{caso contrário} \end{cases}$$

Restrições As seguintes restrições se aplicam ao problema do RWA.

O número de comprimentos de onda em um enlace não deve exceder sua capacidade:

$$\sum_{c \in W} \Lambda_{c,l} \leq |W| \quad , \quad l \in E \quad (5.1)$$

O número de comprimentos de onda usados em um enlace é igual ao número de conexões estabelecidas que usam esse mesmo enlace:

$$\sum_{s \in N} \sum_{d \in N} \sum_{r \in P_{s,d}} V_l^{s,d,r} \times \sigma_{s,d}^r = \sum_{c \in W} \Lambda_{c,l} \quad , \quad l \in E \quad (5.2)$$

O número de conexões estabelecidas para cada par origem-destino (s, d) não pode exceder a demanda:

$$\sum_{r \in P_{s,d}} \sigma_{s,d}^r \leq T_{s,d} \quad , \quad s, d \in N \quad (5.3)$$

Os comprimentos de onda escolhidos devem corresponder ao tráfego alocado a cada par

origem-destino:

$$\sum_{c \in W} \lambda_{c,r}^{s,d} = \sigma_{s,d}^r, \quad s, d \in N, r \in P_{s,d} \quad (5.4)$$

Cada comprimento de onda em um enlace é usado por, no máximo, uma conexão:

$$\sum_{s \in N} \sum_{d \in N} \sum_{r \in P_{s,d}} V_l^{s,d,r} \times \lambda_{c,r}^{s,d} = \Lambda_{c,l}, \quad c \in W, l \in E \quad (5.5)$$

Não há necessidade da restrição de continuidade de comprimentos de onda, pois isto já esta implicitamente implementado pois se fez $\lambda_{c,r}^{s,d}$ independente de l .

Função Objetivo O objetivo é maximizar o número de conexões que podem ser estabelecidas na rede:

$$Z = \sum_{s \in N} \sum_{d \in N} \sum_{r \in P_{s,d}} \sigma_{s,d}^r \quad (5.6)$$

5.2 RWA com Restrições de Potência (RWA-P)

Nesta Seção, tratar-se-á do problema do roteamento e alocação de comprimento de onda considerando-se a restrição de potência (RWA-P) [6, 5], que pode ser definido assim: dados uma topologia de rede e um conjunto de conexões a serem estabelecidas, o objetivo é maximizar o número de conexões atendidas através do roteamento e alocação de comprimentos de onda, mantendo-se um nível adequado de potência óptica e de relação sinal-ruído na rede. Assim, devem-se obedecer às seguintes restrições:

1. Uma conexão é roteada usando um mesmo comprimento de onda em todos os enlaces da rede;
2. Duas conexões não podem compartilhar um mesmo comprimento de onda em um mesmo enlace;

3. A potência do sinal é mantida acima de um limiar (P_{sen}) da origem até o destino. Esse limiar pode ser calculado através das Equações 4.19 e 4.14;
4. A potência total de todos os sinais em um enlace deve ser menor ou igual a P_{max} , onde P_{max} é a potência agregada máxima permitida em um enlace.

O RWA-P também é NP-completo, pois contém o problema do RWA como caso especial. Sua formulação matemática será apresentada a seguir.

5.2.1 Formulação Matemática

O problema pode ser descrito numa formulação mista-inteira não-linear (MINLP) que é mostrada a seguir. Assume-se, sem perda de generalidade, que somente um rota (por exemplo, a mais curta) é considerada.

5.2.1.1 Entradas do problema

O RWA-P tem as seguintes entradas:

- A topologia física da rede, descrita como um grafo bi-direcional $G = (V, E)$, onde V é o conjunto formado por estações e *cross-connects* e E é o conjunto de enlaces da rede. Desse grafo G , é derivado o número de enlaces $L = |E|$.
- O conjunto de estações N da rede ($N \subset V$). Estações são identificadas pelos índices $1, 2, \dots, N$;
- O conjunto de *switches* M da rede ($M \subset V$). *Switches* são indicados pelos índices $(N + 1), (N + 2), \dots, (N + M)$;
- Matriz de demanda $T_{|N| \times |N|}$, o qual especifica as requisições de conexão entre as estações. $T_{s,d}$ é o número de requisições de conexão da estação de origem s à estação de destino d ;

- O conjunto de comprimentos de onda W disponíveis em um enlace. Cada estação tem $|W|$ transmissores e $|W|$ receptores para receber em qualquer um dos $|W|$ comprimentos de onda;
- Origem s_l do enlace l , onde $1 \leq s_l \leq (N + M)$;
- Destino d_l do enlace l , onde $1 \leq d_l \leq (N + M)$;
- Comprimento L_l do enlace l (em km);
- Número de amplificadores ∇_l no enlace l ;
- O comprimento $\eta_{l,k}$ da fibra k no enlace l (em km), $1 \leq k \leq (2\nabla_l + 1)$ e k ímpar;
- O grau D_i de entrada e saída do *switch* i ;
- O conjunto de enlaces OUT_i que tem o *switch* i como origem;
- O conjunto de enlaces IN_i que tem o *switch* i como destino;
- A tabela de roteamento $R_{s,d,l}$ para cada par origem-destino (s, d) . Cada par origem-destino tem uma rota.

$$R_{s,d,l} = \begin{cases} 1 & : \text{se o enlace } l \text{ pertence a rota que vai da origem até o destino} \\ 0 & : \text{caso contrário} \end{cases}$$

5.2.1.2 Variáveis

O RWA-P tem três categorias de variáveis: de roteamento, de enlace e de nó.

Variáveis de roteamento Há um único tipo de variável de roteamento:

- Variável binária $\Lambda_{s,d,c}$ indicando o uso do comprimento de onda c por um caminho óptico qualquer.

$$\Lambda_{s,d,c} = \begin{cases} 1 & : \text{ se o comprimento de onda } c \text{ é usado pela rota definida entre} \\ & \text{a estação de origem } s \text{ e a estação de destino } d \\ 0 & : \text{ caso contrário} \end{cases}$$

Variáveis de enlace Aqui serão mostradas as variáveis indexadas pelos enlaces. É adotada a convenção *wm* que variáveis maiúsculas se referem a níveis de potência agregada no enlace e são medidas em mW. As variáveis minúsculas se referem a níveis de potência por comprimento de onda e são medidas em dBm.

Devido ao fato de haver potências representadas em mW e dBm, é necessário definir duas funções para se fazer a conversão:

- $toDB(\xi) = 10 \log_{10}(\xi)$ - para se converter de mW para dBm.
- $toMW(\xi) = 10^{\xi/10}$ - para se converter de dBm para mW.

Cada enlace da rede é dividido em componentes. Um componente é um dispositivo ou segmento de fibra que altera o nível do sinal de potência, amplificando ou atenuando esse sinal. Um exemplo de componente é o amplificador óptico. Se o enlace não contém nenhum dispositivo, então o enlace possui um único componente que é a fibra. Em geral, se um enlace l tem ∇_l dispositivos, então há $(2\nabla_l + 1)$ componentes, pois cada dispositivo no enlace encontra-se entre dois segmentos de fibra (veja Figura 5.1). Introduz-se essas variáveis para o enlace para se garantir que os níveis de potência de cada sinal óptico estão dentro dos níveis considerados aceitáveis no início e no fim de cada componente de cada enlace da rede. Considera-se nessa formulação o uso dos seguintes dispositivos: amplificadores de linha, pré-amplificadores, amplificadores de potência, conectores, multiplexadores e demultiplexadores. Outros dispositivos podem ser facilmente adicionados ao modelo. As variáveis de enlace são:

- A variável $p_{c,s,d,l,x}^{beg}$ representa a potência do comprimento de onda c para a conexão que tem como par origem-destino (s, d) no início do componente x no enlace l . Essa variável não deve ter valor menor que p_{sen} e nem valor maior que P_{max} .
- A variável $p_{c,s,d,l,x}^{end}$ representa a potência do comprimento de onda c para a conexão que tem como par origem-destino (s, d) no final do componente x no enlace l . Essa variável não deve ter valor menor que p_{sen} e nem valor maior que P_{max} .
- A variável $P_{l,x}^{beg}$ representa a potência agregada no início de cada componente x . Essa variável não deve ter valor menor que p_{sen} e nem valor maior que P_{max} .
- A variável $P_{l,x}^{end}$ representa a potência agregada no final de cada componente x . Essa variável não deve ter valor menor que p_{sen} e nem valor maior que P_{max} .

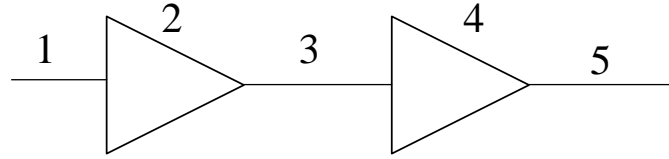


Figura 5.1: Um enlace com 2 EDFAs e 3 segmentos de fibra. Este enlace tem 2 dispositivos e 5 componentes.

Variáveis de nó Há dois tipos de variável de nó:

- A variável $p_{c,s,d,i,l}^{switch_out}$ representa a potência (em dBm) do comprimento de onda c na saída do *switch* i (ligado ao enlace l) para a conexão que tem como par origem-destino (s, d) , onde $(N + 1) \leq i \leq (N + M)$ e o enlace $l \in OUT_i$.
- A variável $p_{c,s,d}^{xmit}$ representa a potência (em dBm) de transmissão da estação de origem s no comprimento de onda c para a conexão que tem como par origem-destino (s, d) , onde $1 \leq s \leq N$.

Variáveis Básicas e Não-Básicas Nesta formulação, há dois tipos de variáveis: básicas e não-básicas. As variáveis básicas são aquelas que são resolvidas pelo *solver* e não podem ser escritas em termos de outras variáveis. No nosso caso, há apenas dois tipos de variáveis básicas: $p_{c,s,d}^{xmit}$ e $\Lambda_{s,d,c}$. Todas as outras variáveis podem ser escritas em termos dessas variáveis e são, dessa forma, não-básicas.

5.2.1.3 Restrições

Há dois grupos de restrições: de roteamento e de potência.

Restrições de Roteamento As seguintes restrições de roteamento se aplicam ao problema do RWA-P.

Um caminho óptico só é estabelecido se ele for requisitado:

$$\sum_{c \in W} \Lambda_{s,d,c} \leq T_{s,d} \quad , \quad 1 \leq s, d \leq N \quad (5.7)$$

Cada enlace não pode ser usado para rotear mais do que W conexões (restrição de capacidade):

$$\sum_{s=1}^N \sum_{d=1}^N \sum_{c \in W} \Lambda_{s,d,c} \times R_{s,d,l} \leq |W| \quad , \quad 1 \leq l \leq L \quad (5.8)$$

Dois caminhos ópticos que compartilham o mesmo enlace l devem usar diferentes comprimentos de onda:

$$\sum_{s=1}^N \sum_{d=1}^N \Lambda_{s,d,c} \times R_{s,d,l} \leq 1 \quad , \quad 1 \leq l \leq L \quad , \quad c \in W \quad (5.9)$$

A variável que indica o uso de um comprimento de onda c é binária:

$$\Lambda_{s,d,c} \in \{0, 1\} \quad (5.10)$$

A restrição de continuidade de comprimento de onda foi implicitamente estabelecida ao se fazer $\Lambda_{s,d,c}$ independente de l .

Restrições de Potência As seguintes restrições de potência se aplicam ao problema do RWA-P.

A potência de transmissão do comprimento de onda c da estação de origem s à estação de destino d deve estar entre níveis de potência considerados aceitáveis:

$$p_{sen} \leq p_{c,s,d}^{xmit} \leq toDB(p_{max}^{xmit}) \quad (5.11)$$

Para o *switch* i , onde o enlace $l \in OUT_i$, temos:

$$p_{c,s,d,i,l}^{switch_out} = p_{c,s,d,l,(2\nabla_l+1)}^{end} - LOSS_i \times \Lambda_{s,d,c}, \quad (5.12)$$

onde $LOSS_i$ é a perda causada pela inserção do *switch* i .

A potência do comprimento de onda c no início do primeiro componente depende da origem do enlace l ser uma estação ou *switch*:

$$p_{c,s,d,l,1}^{beg} = p_{c,s,d}^{xmit} \quad , \quad 1 \leq s_l \leq N \quad (5.13)$$

$$p_{c,s,d,l,1}^{beg} = p_{c,s,d,i,l}^{switch_out} \quad , \quad (N+1) \leq i = s_l \leq (N+M) \quad (5.14)$$

A potência do comprimento de onda c no início do componente k , $k \neq 1$, no enlace l é:

$$p_{c,s,d,l,k}^{beg} = p_{c,s,d,l,k-1}^{end} \quad (5.15)$$

A potência total (em mW) no início do componente k no enlace l é definida como:

$$p_{l,k}^{beg} = \sum_{s=1}^N \sum_{d=1}^N \sum_{c \in W} toMW(p_{c,s,d,l,k}^{beg}) \times \Lambda_{s,d,c} \times R_{s,d,l} \quad (5.16)$$

Similarmente, a potência total (em mW) no fim do componente k no enlace l é:

$$p_{l,k}^{end} = \sum_{s=1}^N \sum_{d=1}^N \sum_{c \in W} toMW(p_{c,s,d,l,k}^{end}) \times \Lambda_{s,d,c} \times R_{s,d,l} \quad (5.17)$$

Definindo a função $\Upsilon_{l,k}(P_{in})$ como a função que retorna o ganho ou perda do componente k no enlace l , dado como entrada a potência agregada que entra nesse componente $P_{in} = P_{l,k}^{beg}$.

Assim:

$$p_{c,s,d,l,k}^{end} = p_{c,s,d,l,k}^{beg} + \Upsilon_{l,k}(P_{in}) \quad (5.18)$$

Os níveis de potência para cada comprimento de onda no início de cada componente x devem estar entre níveis considerados aceitáveis:

$$p_{sen} \leq p_{c,s,d,l,x}^{beg} \times \Lambda_{s,d,c} \leq toDB(P_{max}) \quad (5.19)$$

Similarmente, os níveis de potência para cada comprimento de onda no fim de cada componente x devem estar entre níveis considerados aceitáveis:

$$p_{sen} \leq p_{c,s,d,l,x}^{end} \times \Lambda_{s,d,c} \leq toDB(P_{max}) \quad (5.20)$$

A potência total no início e no fim de cada componente x , $x \neq 1$, é a soma de todas as potências individuais depois de mudanças de escala e devem estar entre níveis considerados aceitáveis:

$$toMW(p_{sen}) \leq P_{l,k}^{beg} \leq P_{max} \quad (5.21)$$

$$toMW(p_{sen}) \leq P_{l,k}^{end} \leq P_{max} \quad (5.22)$$

5.2.1.4 Função Objetivo

O objetivo é maximizar o número de conexões que podem ser estabelecidas na rede:

$$Z_1 = \sum_{s=1}^N \sum_{d=1}^N \sum_{c \in W} \Lambda_{s,d,c} \quad (5.23)$$

5.2.2 Solução em Duas Fases

O problema do RWA com restrições de potência (RWA-P) é um problema misto-inteiro não-linear. Um programa é considerado não-linear quando ele possui uma ou mais restrições que não podem ser escritas na forma $aX + b \leq c$ e/ou a função objetivo não pode ser escrita na forma $aX + b$, onde a , b e c são constantes. Embora essa formulação tenha uma função objetivo linear (veja Equação 5.23), ela possui restrições não-lineares (veja as Equações 5.16, 5.17, 5.19 e 5.20). Além disso, o modelo de ganho para o EDFA também não é linear. Finalmente, há uma restrição de integralidade na Equação 5.10 que complica o problema, necessitando do uso de técnicas inteligentes para se restringir o espaço de busca.

Claramente, essa formulação do RWA-P resulta em um problema muito complexo e de difícil solução. Para contornar o problema da não-linearidade, o problema do RWA-P deve ser dividido em dois sub-problemas. Primeiramente, se resolve o RWA sem se considerar as restrições de potência, como mostrado na Sub-seção 5.1.2. Pode-se usar o programa ILOG CPLEX [1] para se resolver o problema do RWA, pois ele é linear. Obtém-se da resolução do RWA um conjunto de caminhos ópticos que atendem a um dado número de requisições de conexão.

A partir desses caminhos ópticos obtidos, tenta-se estabelecer as conexões no que diz respeito às restrições de potência, atribuindo-se potências de transmissão para as estações de origem.

Duas heurísticas são sugeridas para se fazer essa atribuição de potência, respeitando-se as restrições de potência: os algoritmos genéticos e a heurística Menor Ganho Primeiro (*Smallest-Gain-First* – SGF). Elas usam mecanismos comuns de estabelecimento de conexões, mas diferem quanto ao modo de atribuir os valores de transmissão para as estações de origem.

5.2.2.1 Mecanismo de Estabelecimento de Conexão

Estabelecer uma conexão significa que uma potência de transmissão foi calculada para a estação de origem dessa conexão de tal forma que as quatro restrições presentes na definição do RWA-P (Seção 5.2) são atendidas. A restrição 3 leva em conta, indiretamente, o ruído ASE dos amplificadores ópticos, pois esse ruído é considerado pelas Equações 4.5 e 4.14, como descrito na Sub-seção 4.1.3.

A ordem com que se tenta o estabelecimento das conexões é importante: quando se tenta estabelecer uma conexão i , essa conexão pode causar um decréscimo do ganho dos amplificadores ópticos por onde ela passa ou um aumento do nível de ruído, fazendo com que conexões previamente estabelecidas tenham seu nível de potência fora dos limites considerados aceitáveis. Assim, a conexão i deve ser rejeitada.

Para maiores detalhes sobre esses mecanismos, consulte o Apêndice A.

5.2.2.2 Algoritmos Genéticos

Os algoritmos genéticos (GA) são uma classe de métodos estocásticos de otimização que simulam o processo de evolução da Natureza (veja o Capítulo 2 para maiores informações). Os algoritmos genéticos são métodos de uso geral que apresentam uma grande robustez para a otimização de problemas difíceis, como o RWA-P.

Para se resolver o RWA-P utilizando-se algoritmos genéticos é necessário definir:

- A representação genética dos indivíduos. Cada gene determina a potência de transmissão de uma conexão, que é do tipo real $\in [0, P_{max}^{xmit}]$. A ordem dos genes está

relacionada com a ordem da conexão, isto é, a primeira conexão é representada pelo primeiro gene e assim sucessivamente.

- A criação da população inicial, que é feita de forma aleatória, a não ser um indivíduo dessa população que é definido usando uma heurística semelhante a SGF, mas que não reordena as rotas;
- A função objetivo, que é o algoritmo 1 (que pode ser encontrado no Apêndice A);

Além disso, é necessário definir os operadores genéticos e os parâmetros dos algoritmos. Isto é feito na Seção 6.3 para cada modelo de rede óptica considerado.

5.2.2.3 Heurística Menor-Ganho-Primeiro (*Smallest-Gain-First* – SGF)

Dadas as m conexões obtidas resolvendo-se o problema do RWA, as conexões são ordenadas segundo a Equação 5.24, a qual calcula a quantidade total de ganho ou perda que a conexão i encontra ao longo do seu percurso, assumindo-se que nenhuma outra conexão está presente na rede.

A heurística SGF assume que os amplificadores ópticos estão uniformemente distribuídos nos enlaces e tenta estabelecer primeiramente as conexões que requerem menos recursos da rede, ou seja, as conexões que terão menor impacto no estabelecimento das próximas conexões, que são as que têm menor ganho ou maior perda.

$$g_i = \sum_{l \in \text{path}[i]} \left(LOSS_{d_l} + \sum_{c \in l} \Upsilon_{l,c}(0) \right), \quad (5.24)$$

onde $\Upsilon_{l,c}(0)$ denota a perda/ganho do componente c no enlace l assumindo-se que nenhuma outra conexão entra no mesmo componente (logo, $P_{in} = 0$ mW); d_l é o nó de destino do enlace l ; e $LOSS_{d_l}$ é a perda de inserção do *switch* que se encontra no destino do enlace l , de forma que essa perda só é calculada se $(N + 1) \leq d_l \leq (N + M)$.

Nesta heurística, se na conexão sendo estabelecida o nível de potência fica menor que p_{sen} , então essa conexão é terminada, a potência de transmissão é incrementada de um δ e o processo de estabelecimento de conexão é repetido até que a conexão seja estabelecida ou que uma violação de potência de outro tipo aconteça (por exemplo, a potência da conexão fique maior que P_{max} em algum componente).

Capítulo 6

Simulação e Resultados

6.1 Parâmetros

Parâmetro	Valor
Máxima potência agregada no enlace (P_{max})	1 mW (0 dBm)
Potência máxima de transmissão (P_{max}^{xmit})	1 mW (0 dBm)
Ganho de pequeno sinal do amplificador de linha (G_{inline})	20 dB
Ganho de pequeno sinal do pré-amplificador (G_{input})	12 dB
Ganho de pequeno sinal do amplificador de potência (G_{output})	12 dB
Atenuação da fibra (α)	0,2 db/km
Perda devida ao conector (L_{tap})	1 dB
Perda devida ao multiplexador (L_{mx})	4 dB
Perda devida ao demultiplexador (L_{dm})	4 dB
Perda devida à inserção do <i>switch</i> (L_s)	1 dB
Perda devida ao acoplamento do <i>switch</i> (L_w)	1 dB
Frequência da portadora óptica (f_c)	193 THz (1,55 μ m)
Constante de Planck (h)	$6,63 \times 10^{-34}$ J/Hz
Faixa óptica (B_o)	100 GHz
Taxa de bits	2,5 Gbps
Faixa elétrica (B_e)	2 GHz
Fator de emissão espontânea do amplificador de linha (N_{sp}^{inline})	2
Fator de emissão espontânea do pré-amplificador (N_{sp}^{input})	2
Fator de emissão espontânea do amplificador de potência (N_{sp}^{output})	2
Potência de saturação do amplificador de linha (P_{inline}^{sat})	13,7 mW
Potência de saturação do pré-amplificador de linha (P_{input}^{sat})	13,7 mW
Potência de saturação do amplificador de potência (P_{output}^{sat})	13,7 mW
Parâmetro γ da função de Q (probabilidade de erro)	8

Tabela 6.1: Parâmetros e seus respectivos valores usados no RWA-P.

6.2 Arquitetura do Nó e da Rede

Uma rede óptica consiste na interconexão de nós capazes de rotear comprimentos de onda (*Wavelength-Routing Nodes* – WRN). Um par de fibras uni-direcionais conectam WRNs na rede. Um WRN contém componentes tais como conectores, amplificadores, multiplexadores, etc. Além disso, as estações locais e o *cross-connect* são considerados parte do WRN.

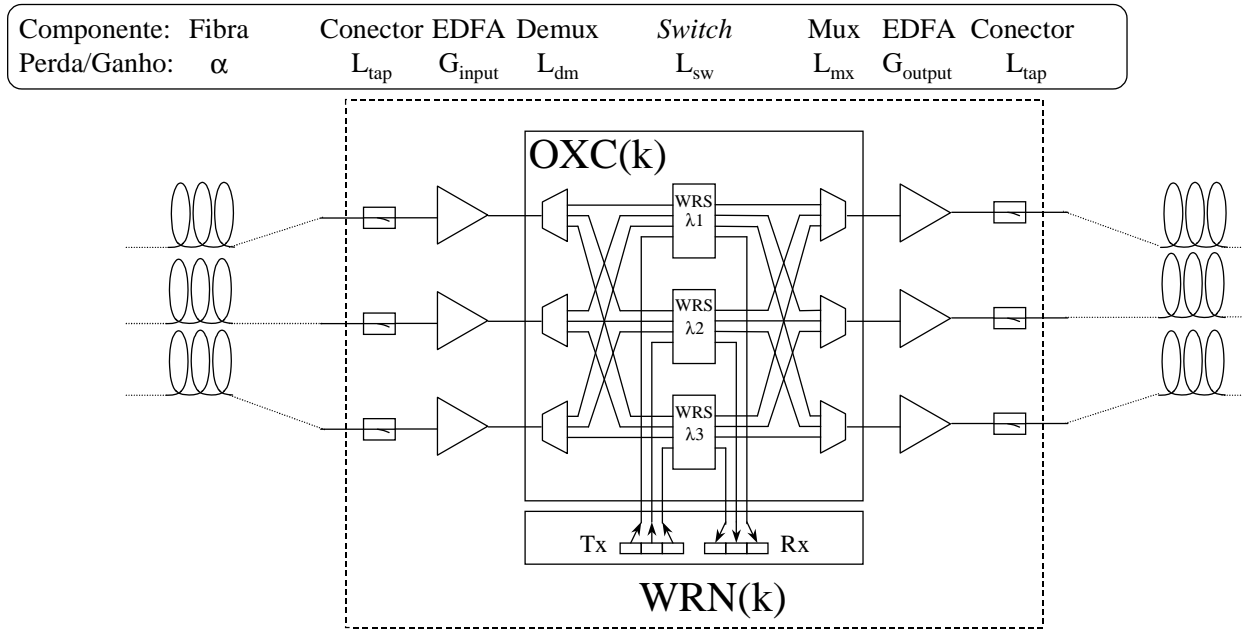


Figura 6.1: Arquitetura do WRN.

A Figura 6.1 mostra um WRN típico. Este WRN 4×4 tem uma única estação local representada pelos receptores e transmissores. O WRN também contém um *cross-connect* óptico (OXC). O OXC utiliza multiplexadores e demultiplexadores e comutadores de comprimento de onda (*Wavelength Routing Switch* – WRS) para realizar todos os tipos de permutação de tráfego.

Neste exemplo, há 3 WRS no nó e cada WRS é dedicado a um comprimento de onda, ou seja, esse nó somente suporta 3 comprimentos de onda. O sinal entrando no WRN encontra vários componentes que contribuem ou para o ganho ou para a perda do sinal. A perda

causada por um WRS $D_i \times D_i$ é dada pela seguinte fórmula [44]:

$$L_{sw} = 2\lceil \log_2(D_i) \rceil L_s + 4L_w, \quad (6.1)$$

onde D_i é o grau do nó, ou seja, o número de enlaces e estações que são ligadas a esse nó; L_s é a perda devida à inserção e L_w é a perda devida ao acoplamento. Como dito anteriormente, tanto o OXC como o WRS são referidos simplesmente como *switch* neste trabalho.

6.3 Os Modelos de Redes Usados

Para esta Dissertação trabalhou-se com dois modelos diferentes de rede: uma rede exemplo de pequenas dimensões e a Rede Italiana de Alta Velocidade [5].

A Tabela 6.2 mostra todos os métodos e operadores genéticos pesquisados nesta Dissertação.

Método de Seleção	Roulette Wheel [9] Stochastic Universal Sampling [9] Rank Selection [49] Tournament Selection [20]
Operador de Recombinação	Simulated Binary Crossover (SBX) [15] Blend Crossover (BLX-0.5) [18] Arithmetical Crossover [30]
Operador de Mutação	Non-uniform Mutation [30] Cauchy Mutation [50] Gaussian Mutation [17] Logarithmic Mutation [23]
Estratégia de Reinscrição	<i>Overlapping</i> e elitista

Tabela 6.2: Métodos e operadores genéticos testados.

Após extensa busca pelos melhores parâmetros, métodos e operadores genéticos, adotou-se a Tabela 6.3 na simulação dos dois modelos de rede. As simulações foram feitas em um computador Pentium IV de 2 GHz, usando o JDK 1.4.2.

Método de Seleção	<i>Rank Selection</i> com ranking não-linear [7]
Operador de Recombinação	<i>Simulated Binary Crossover</i> [15]
Operador de Mutação	<i>Non-Uniform Mutation</i> [30]
Estratégia de Reinserção	<i>Overlapping</i> e elitista
Tamanho da População	30
Tamanho do <i>pool</i> de Seleção	10
Probabilidade de Recombinação p_c	0,9
Probabilidade de Mutação p_m	0,05

Tabela 6.3: Parâmetros, métodos e operadores genéticos usados nas simulações.

6.3.1 Rede Exemplo de Pequenas Dimensões

Considere a rede de pequenas dimensões mostrada na Figura 6.2. As Tabelas 6.4, 6.5 e 6.6 mostram os resultados obtidos quando se varia o número de comprimentos de onda disponíveis e o número de caminhos mais curtos que devem ser usados. Foi usada a matriz de demanda T (6.2), onde $T_{i,j}$ é o número de conexões requisitadas da estação i à estação j . No total, há 39 requisições de conexão. Essa matriz é a mesma usada em [6, 5].

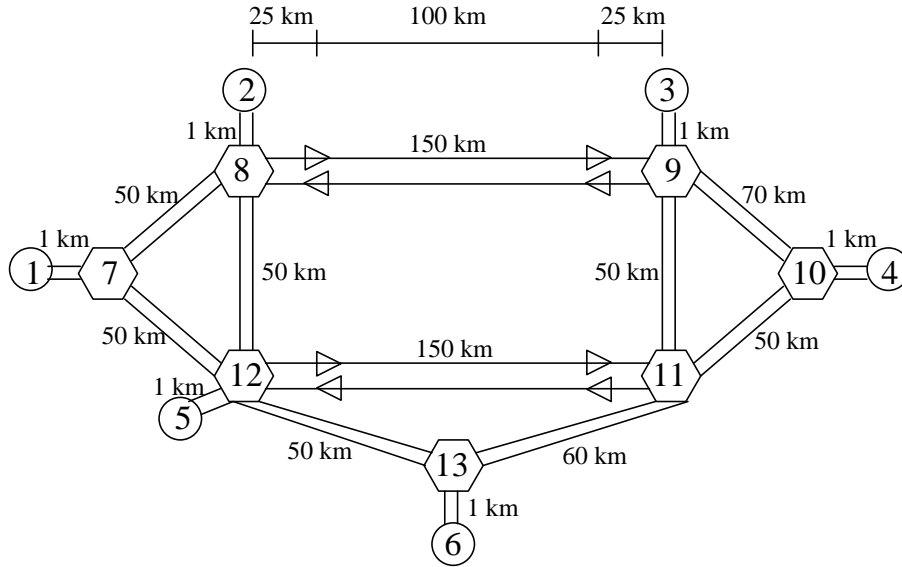


Figura 6.2: Rede exemplo.

$$T = \begin{bmatrix} 0 & 3 & 2 & 0 & 2 & 1 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 3 \\ 1 & 3 & 2 & 0 & 3 & 2 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 1 & 2 & 3 & 1 & 2 & 0 \end{bmatrix} \quad (6.2)$$

As tabelas de resultado são construídas da seguinte forma: a primeira coluna mostra o número W de comprimentos de onda disponíveis em um enlace. A segunda coluna se refere ao número K de caminhos mais curtos usados pelo RWA lógico. A terceira coluna dá o número de conexões geradas pela resolução do RWA topológico (que serve como um limitante superior para o número de conexões que podem ser estabelecidas pelas heurísticas). A quarta coluna indica o número de conexões estabelecidas pela heurística SGF e a quinta coluna indica o tempo de execução da heurística SGF. E, finalmente, a sexta coluna mostra o número de conexões estabelecidas pelo algoritmo genético após 400 gerações e a sétima coluna indica o tempo de execução do algoritmo genético. Foram usadas 400 gerações, pois o algoritmo genético já convergiu por volta desse número de gerações em todos os casos.

Primeiramente, se simulou esta rede usando o modelo de ganho do amplificador descrito em [6, 5], que é dado pela Equação 6.3, com o nível de sensibilidade P_{sen} fixo em -30 dBm, obtendo os resultados mostrados na Tabela 6.4. Os resultados obtidos são muito semelhantes aos obtidos por Ali em [6, 5], de forma que é possível comparar esses resultados com aqueles obtidos pelo modelo proposto neste trabalho.

$$G = \min(SSG, (P_{max}/P_{in})) \quad (6.3)$$

, onde P_{in} é a potência total de entrada (em dBm), P_{max} é a potência máxima de saída (em dBm) e SSG é o ganho de pequeno sinal do amplificador.

Para se verificar a mudança introduzida pelo cálculo da sensibilidade usando as Equações 4.14 e 4.19, a simulação foi repetida com os mesmos parâmetros e obteve-se a Tabela 6.5.

W	K	ILP	SGF	tempo	GA	tempo
7	1	31	29	1 min 47 s	31	21 min 44 s
8	1	33	28	2 min 12 s	30	23 min 53 s
9	1	35	29	2 min 29 s	31	27 min 50 s

Tabela 6.4: Conexões estabelecidas na rede exemplo com sensibilidade fixa.

W	K	ILP	SGF	tempo	GA	tempo
7	1	31	26	1 min 12 s	28	15 min 47 s
	2	32	17	8 min 05 s	23	23 min 11 s
8	1	33	27	2 min 53 s	30	18 min 52 s
	2	35	20	4 min 24 s	25	24 min 41 s
9	1	35	28	4 min 09 s	30	26 min 33 s
	2	37	21	15 min 39 s	23	27 min 55 s

Tabela 6.5: Conexões estabelecidas na rede exemplo com sensibilidade calculada.

Finalmente, se introduz o modelo de ganho do amplificador da Sub-seção 4.1.1 na simulação anterior, obtendo-se os resultados da Tabela 6.6.

W	K	ILP	SGF	tempo	GA	tempo
7	1	31	7	2 min 50 s	19	14 min 29 s
	2	32	7	8 min 48 s	19	21 min 17 s
8	1	33	7	5 min 09 s	24	21 min 17 s
	2	35	6	4 min 39 s	17	23 min 57 s
9	1	35	7	5 min 49 s	25	24 min 50 s
	2	37	7	16 min 24 s	19	21 min 44 s

Tabela 6.6: Conexões estabelecidas na rede exemplo com sensibilidade calculada e modelo de amplificador melhorado.

Na Figura 6.3, é mostrada a convergência do algoritmo genético para o caso com sensibilidade calculada e modelo de amplificador melhorado.

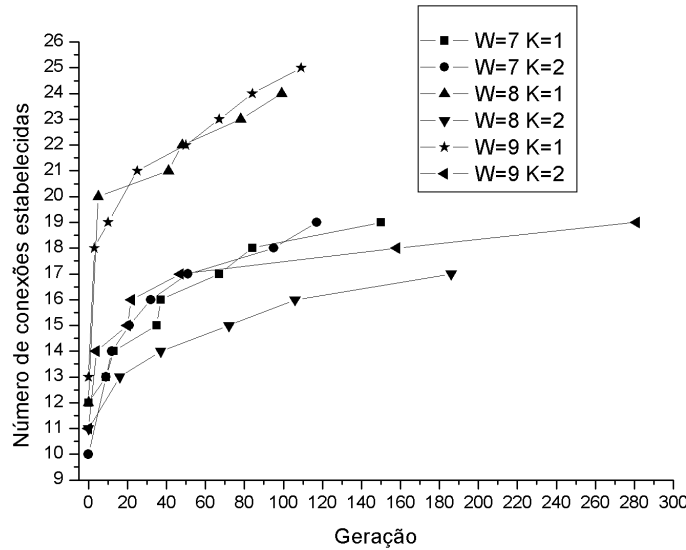


Figura 6.3: Convergência do algoritmo genético para a rede exemplo com sensibilidade calculada e modelo de amplificador melhorado.

6.3.2 Rede Italiana de Alta Velocidade

A rede italiana de alta velocidade é uma rede transparente de grandes dimensões e foi escolhida por ser considerada representativa das futuras redes totalmente ópticas e roteadas por comprimento de onda. Possui 21 estações e 114 enlaces unidirecionais. Sua topologia pode ser vista na Figura 6.4, a qual é explicada com detalhes no Apêndice B.

A matriz de tráfego para a rede italiana é dada pela Tabela 6.7. No total há 54 requisições de conexão. Essa matriz é a mesma usada em [6, 5].

As tabelas de resultados são organizadas da seguinte forma: a primeira coluna mostra o número W de comprimentos de onda disponíveis em um enlace. A segunda coluna se refere ao número K de caminhos mais curtos usados pelo RWA topológico. A terceira coluna dá o número de conexões geradas pela resolução do RWA topológico (que serve como um limitante superior para o número de conexões que podem ser estabelecidas pelas heurísticas). A quarta coluna indica o número de conexões estabelecidas pela heurística SGF e a quinta

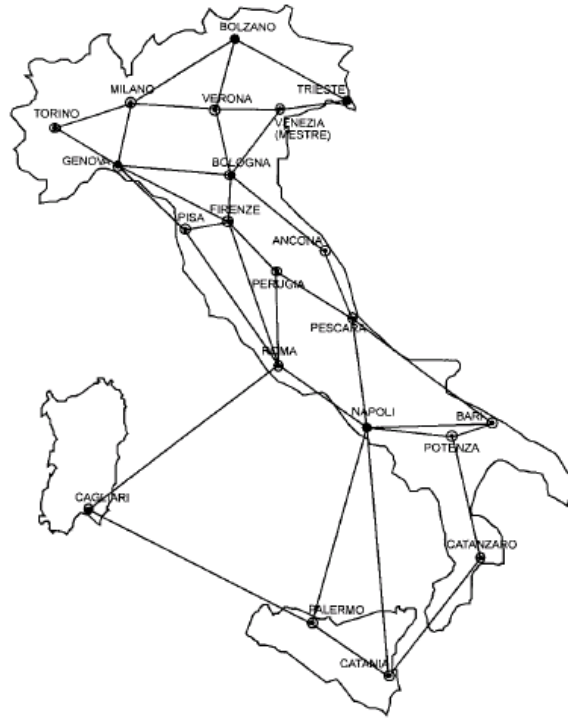


Figura 6.4: Rede italiana de alta velocidade.

coluna indica o tempo de execução da heurística SGF. E, finalmente, a sexta coluna mostra o número de conexões estabelecidas pelo algoritmo genético após 400 gerações e a sétima coluna indica o tempo de execução do algoritmo genético. Foram usadas 400 gerações, pois o algoritmo genético já convergiu por volta desse número de gerações em todos os casos.

De forma similar à rede exemplo da Sub-seção anterior, a Rede Italiana foi simulada em três diferentes situações, conforme mostrado a seguir.

Primeiramente, se simulou esta rede usando o modelo de ganho do amplificador descrito em [6, 5], que é dado pela Equação 6.3, com o nível de sensibilidade P_{sen} fixo em -30 dBm, obtendo os resultados mostrados na Tabela 6.8.

Para se verificar a mudança introduzida pelo cálculo da sensibilidade usando as Equações 4.14 e 4.19, a simulação foi repetida com os mesmos parâmetros e obteve-se a Tabela 6.9.

Na Figura 6.5, é mostrada a convergência do algoritmo genético para o caso com sensibi-

	Bolzano	Torino	Milano	Verona	Venezia	Trieste	Genova	Bolzano	Pisa	Firenze	Ancona	Perugia	Roma	Pescara	Cagliari	Napoli	Bari	Potenza	Palermo	Catanzaro	Catania
Bolzano	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Torino	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0
Milano	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Verona	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
Venezia	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Trieste	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
Genova	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
Bolzano	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
Pisa	0	1	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1
Firenze	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0
Ancona	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Perugia	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0
Roma	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Pescara	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
Cagliari	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Napoli	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Bari	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
Potenza	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0
Palermo	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
Catanzaro	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
Catania	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Tabela 6.7: Matriz de tráfego para a rede italiana.

W	K	ILP	SGF	tempo	GA	tempo
7	1	52	19	12 min 01 s	27	29 min 01 s
8	1	53	19	12 min 55 s	29	28 min 57 s
9	1	54	19	13 min 28 s	27	28 min 03 s

Tabela 6.8: Conexões estabelecidas na Rede Italiana com sensibilidade fixa

W	K	ILP	SGF	tempo	GA	tempo
7	1	52	18	15 min 02 s	27	29 min 48 s
8	1	53	21	17 min 18 s	28	30 min 10 s
9	1	54	21	17 min 13 s	28	37 min 02 s

Tabela 6.9: Conexões estabelecidas na Rede Italiana com sensibilidade calculada.

lidade calculada (sem o uso do modelo de amplificador melhorado).

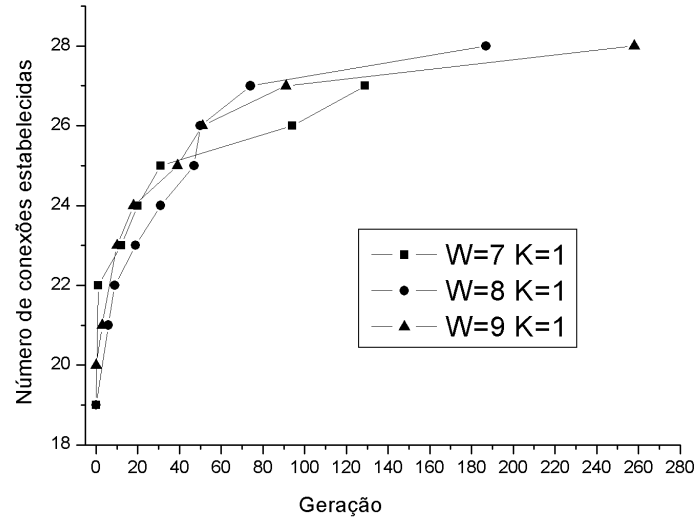


Figura 6.5: Convergência do algoritmo genético para a Rede Italiana com sensibilidade calculada.

Finalmente, se introduz o modelo de ganho do amplificador da Sub-seção 4.1.1 na simulação anterior, obtendo-se os resultados da Tabela 6.10.

W	K	ILP	SGF	tempo	GA	tempo
7	1	52	15	3 min 57 s	15	24 min 33 s
8	1	53	15	4 min 07 s	15	25 min 19 s
9	1	54	15	4 min 08 s	15	25 min 49 s

Tabela 6.10: Conexões estabelecidas na Rede Italiana com sensibilidade calculada e modelo de amplificador melhorado.

Capítulo 7

Conclusão

7.1 Sensibilidade e Ruído ASE

Nesta Dissertação um método para se calcular a sensibilidade em cada componente da rede é proposto, considerando o ruído ASE gerado pelos amplificadores ópticos.

Em [45], a ASE é considerada no roteamento, mas somente através de um cálculo simples que dá o limitante superior para o número de amplificadores ópticos em um caminho óptico, representado pela Equação 7.1.

$$M \leq \left\lfloor \frac{P_L}{2SNR_{min}n_{sp}h\nu(G-1)B_o} \right\rfloor, \quad (7.1)$$

onde M é o número de amplificadores ópticos, P_L é a potência média transmitida, n_{sp} é o fator de emissão espontânea, h é a constante de Planck, ν é a frequência da portadora óptica, B_o é a faixa óptica, G é o ganho do amplificador e SNR_{min} é a relação sinal-ruído mínima.

Em [6, 5], a sensibilidade é fixada em um valor arbitrário e não leva em consideração nenhum tipo de ruído na rede. Além disso, se considera a mesma sensibilidade em qualquer ponto da rede, ignorando-se o número de amplificadores ópticos atravessados por cada caminho óptico.

Comparando-se as Tabelas 6.4 e 6.5 para a rede exemplo, pode-se verificar que com a

sensibilidade fixa no valor de -30 dBm, conexões que deveriam ser bloqueadas por não ter um nível de BER adequado são aceitas e estabelecidas. Por outro lado, comparando-se as Tabelas 6.8 e 6.9 pode-se verificar que, em alguns casos, a sensibilidade fixa causa o bloqueio de conexões que podem ser estabelecidas com uma BER menor que o limite aceitável.

Dessa forma, como o método proposto é um modelo mais sofisticado por considerar todos os componentes da rede, ele fornece um comportamento mais próximo da realidade.

7.2 Algoritmo Genético

O algoritmo genético demonstrou ser uma excelente opção para otimizar o RWA-P. Em alguns casos, como pode ser verificado nas Tabelas 6.4, 6.5 da rede exemplo, o algoritmo genético foi capaz de uma pequena melhora sobre a heurística SGF.

Entretanto, no caso da Tabela 6.6 da rede exemplo nota-se um grande aumento das conexões estabelecidas pelo algoritmo genético em relação à SGF. Isso mostra a eficiência do algoritmo genético, por se tratar do caso mais “complicado” da rede exemplo, pois há presença do cálculo de sensibilidade por componente e de um modelo melhorado para o ganho do amplificador.

Essa eficiência também é demonstrada nas Tabelas 6.8 e 6.9 da rede italiana, onde há um grande aumento das conexões estabelecidas pelo algoritmo genético em relação à SGF. É importante ressaltar que a rede italiana tem grandes dimensões e é altamente conectada. Esse fato aumenta em muito o espaço de busca, justificando assim o uso dos algoritmos genéticos, pois estes fazem um bom aproveitamento do espaço de busca.

Como o algoritmo genético usado para o RWA-P é inicializado com uma solução dada por uma heurística, ele terá, no mínimo, um resultado igual a solução dada pela heurística, como pode ser observado na Tabela 6.10. Contudo, como visto anteriormente, vale a pena gastar um pouco mais de tempo no processamento do algoritmo genético, visando obter soluções

melhores que a obtida pela SGF.

É também importante notar que o algoritmo genético apresentou robustez para as diferentes redes, mesmo usando os mesmos parâmetros, pois se obteve bons resultados e com rápida convergência, como se pode observar na Tabela 7.1. Embora a robustez seja inerente aos algoritmos genéticos, o uso do operador *Simulated Binary Crossover* permite um aumento significativo da auto-adaptabilidade do algoritmo genético para diferentes redes [10].

Geração	Número de conexões estabelecidas
1	12
5	20
41	21
47	22
77	23
99	24

Tabela 7.1: Convergência do algoritmo genético para a rede exemplo com sensibilidade calculada, modelo de amplificador melhorado, $W = 8$ e $K = 1$.

Finalmente, outra característica interessante do algoritmo genético é que, em geral, o tempo de execução é polinomial com o tamanho da instância. Isso é muito importante, visto que o RWA-P é um problema do tipo NP, o qual tem tempo de execução fatorial com o tamanho da instância. Temos que o tempo médio de execução do algoritmo genético para a rede exemplo é de 22 minutos e 32 segundos e para a rede italiana é de 28 minutos e 45 segundos, até se alcançar 400 gerações.

7.3 Solução em Duas Fases e Uso do Caminho Mais Curto

Comparando-se as Tabelas 6.4 e 6.5 para a rede exemplo, verifica-se que o uso de um caminho que não seja mais curto na resolução do RWA lógico produz uma piora significativa na fase de estabelecimento de conexões com restrições de potência.

Isto é devido ao fato de se dividir a solução do RWA-P em duas fases. Não necessariamente algo que introduz uma melhora na primeira fase (o uso de rotas que não sejam a mais curta aumenta o número de conexões atendidas) causa também uma melhora do resultado na segunda fase. Na verdade, o uso de rotas mais longas aumenta o ruído no caminho óptico (devido ao maior número de amplificadores ópticos) e a interação entre rotas distintas (o que causa uma maior saturação dos amplificadores).

7.4 Amplificadores Ópticos

O ganho do amplificador óptico é dependente da saturação, tornando o ganho uma grandeza não determinística e dependente do tráfego [37].

Comparando-se as Tabelas 6.5 e 6.6 na rede exemplo nota-se que o número de conexões estabelecidas também é dependente do modelo de ganho usado para o EDFA. Isso também pode ser verificado na Tabela 6.10, onde o modelo de amplificador melhorado reduziu quase à metade as conexões estabelecidas pelo algoritmo genético em relação à Tabela 6.9.

7.5 Trabalho futuro

A questão do provisionamento de serviços é uma tarefa importante no gerenciamento de redes. Em uma rede WDM, o roteamento e alocação de comprimentos de onda (RWA) é um exemplo de provisionamento de recursos.

Esquemas centralizados para o RWA dinâmico não são apropriados quando se busca escalabilidade e tolerância a falhas em uma rede. Por isso, esquemas distribuídos são a melhor opção nesses casos.

O RWA é um problema do tipo NP-completo, a não ser para topologias em anéis, conforme demonstrado por Waldman et al. em [48]. Como os algoritmos genéticos são um método que pode ser usado para resolver esse tipo de problema [36, 43] e suas variações [5], eles podem

ser incorporados à inteligência de agentes móveis para se obter uma arquitetura distribuída de RWA.

Outra forma de se tratar o problema do RWA é com o uso de Otimização por Colônia de Formigas (*Ant Colony Optimization* – ACO) [16] (o qual por sua vez também se vale de agentes móveis). De fato, já é possível encontrar na literatura o uso de ACO para o roteamento e alocação de comprimentos de onda para o caso estático [47] e para o caso dinâmico [40, 41, 21].

O termo ACO é usado para se referir à classe de algoritmos que foram inspirados no processo de procura e coleta de alimentos por formigas para a otimização de problemas de difícil solução. Entre suas aplicações pode-se citar o problema do caixeiro viajante e problemas de roteamento em geral.

As idéias básicas da ACO são [12]:

- O uso de simulações concorrentes executadas por agentes móveis chamados de formigas para gerar novas soluções para o problema;
- O uso de agentes para a busca local estocástica de forma a se construir as soluções de uma forma incremental;
- O uso de informações coletadas durante simulações anteriores para conduzir a busca para melhores soluções.

Na ACO, através de um processo iterativo, cada formiga constrói uma solução usando dois tipos de informações locais: informação específica do problema e informação adicionada pelas formigas durante iterações anteriores do algoritmo. De fato, enquanto calcula a solução, cada formiga coleta informações sobre as características do problema a ser resolvido e sobre a sua própria performance e utiliza essa informação para modificar a representação do problema, conforme visto localmente por outras formigas. Dessa maneira, a representação do problema

é modificada de tal forma que a informação contida nas soluções obtidas anteriormente podem ser exploradas para a obtenção de soluções melhoradas.

A ACO é usada extensivamente em problemas de roteamento para redes de comunicação. Entre as técnicas de ACO usadas nesse caso, é possível citar AntNet [12], Ants-Routing [46] and Accelerated Ants Routing [28]. AntNet, por exemplo, provou ser superior a seis algoritmos de roteamento no estado da arte em redes Internet.

De fato, distribuir a inteligência pela rede com o uso de agentes móveis possibilita um uso mais flexível dos recursos da rede, pois se pode incorporar ao problema de roteamento características e restrições de qualidade de serviço (QoS) no nível da aplicação ([29],[34]). Outro aspecto importante que deve ser ressaltado, é que os agentes móveis podem ter acesso às restrições impostas pela camada física ao roteamento ([45],[5]), como efeitos não-lineares na transmissão. Tais restrições têm uma dinâmica muito rápida, impossibilitando sua divulgação em tempo hábil para todos os nós da rede.

Assim, o estudo desses temas constituem uma interessante linha de pesquisa para o futuro.

Apêndice A

Algoritmos de Estabelecimento de Conexões

Os algoritmos a seguir foram usados nesta Dissertação tanto para a heurística SGF quanto para os algoritmos genéticos.

Para se estabelecer as conexões é necessário uma tabela $Con[l, c]$, que para cada componente c e cada enlace l da rede possui a lista de todas as conexões que entram nesse componente com seus respectivos níveis de potência.

O algoritmo 1 adiciona as conexões uma a uma, da primeira à última do conjunto. Tem como entrada \mathcal{C} , que é o conjunto de caminhos ópticos estabelecidos pela resolução do RWA, que só considera o aspecto topológico da rede. E tem como saída o número de conexões estabelecidas. Este algoritmo é a função objetivo do algoritmo genético e também é usado pela heurística SGF.

O algoritmo 2 é usado para estabelecer uma conexão. Para se estabelecer as conexões é necessário uma tabela $Con[l, c]$, que para cada componente c e cada enlace l da rede possui a lista de todas as conexões que entram nesse componente com seus respectivos níveis de potência na rede óptica. Tem como entrada o identificador i da conexão, a potência de transmissão $inputPower$ e a variável binária *incremental* que indica se o algoritmo deve procurar um novo valor de potência de transmissão, caso o valor fornecido seja insuficiente para estabelecer a conexão. Assim, *incremental* é verdadeiro para a heurística SGF e falso para

o algoritmo genético. Este algoritmo retorna verdadeiro se a conexão pode ser estabelecida e falso caso contrário.

Este algoritmo começa atribuindo *inputPower* ao primeiro componente da conexão *i* e depois faz uso do algoritmo 3, que recursivamente vai atribuindo os valores de potência no início de cada componente. Após essa atribuição, se verifica com o algoritmo 4 se a conexão *i* viola alguma restrição ou se as conexões previamente estabelecidas passam a violar alguma restrição, dada a saturação dos EDFAs e um maior nível de ASE na rede.

O algoritmo 3 propaga a potência de todas as conexões no enlace *l*, começando do componente 1 até o componente $(2\Delta_l + 1)$. Tem como entrada o identificador *i* da conexão e o enlace *l* onde deve fazer a propagação de potência. Recursivamente, propaga as potências de todas as conexões que compartilham algum componente com a conexão *i* de forma a assegurar a interação entre as conexões já estabelecidas.

O algoritmo 4 é usado para verificar se uma conexão não viola as restrições de potência. Tem como entrada o indentificador *i* da conexão e tem como saída quatro possíveis códigos: OK quando a conexão não viola nenhuma restrição; BELOW_SENSITIVITY quando o nível de potência da conexão é menor que o mínimo permitido; OVER_MAXIMUM quando o nível de potência da conexão é maior que o máximo permitido; e OVERLOADED quando o nível de potência agregado em um componente é maior que o máximo permitido. Este algoritmo faz uso do algoritmo 7 para atribuir os valores de potência mínima de cada componente.

O algoritmo 5 é usado para finalizar uma conexão na rede que não pode ser estabelecida devido à violação de alguma restrição de potência. Tem como entrada o indentificador *i*. Este algoritmo faz uso do algoritmo 6 que é executado recursivamente.

O algoritmo 6 vai recursivamente removendo qualquer referência à conexão *i* e propaga os novos valores de potência de cada componente de forma semelhante ao algoritmo 3. Tem como entrada o identificador *i* da conexão e o enlace *l* onde deve fazer a nova propagação de potência.

O algoritmo 7 calcula e atribui os valores de potência mínima para cada componente por onde passa a conexão i . Tem como entrada o identificador i de conexão.

A.1 Listagens

Algoritmo 1 EVALUATE(\mathcal{C})

```
Number  $\leftarrow$  0  
 $k \leftarrow |\mathcal{C}|$   
for  $i \leftarrow 1$  to  $k$  do  
  if ADDCONNECTION( $\mathcal{C}_i$ ) then  
    Number  $\leftarrow$  Number + 1  
  end if  
end for  
/* Retorna o número de conexões estabelecidas */  
return Number
```

Algoritmo 2 ADDCONNECTION($i, inputPower, incremental$)

```

 $f \leftarrow$  o primeiro enlace da conexão  $i$ 
if  $i \notin Con[f, 1]$  then
    Adicionar a conexão  $i$  em  $Con[f, 1]$  com potência igual a  $inputPower$ 
else
    A conexão  $i$  em  $Con[f, 1]$  é agora igual a  $inputPower$ 
end if
ADDCONNETIONREC( $i, f$ )
 $check \leftarrow$  CHECKTHISCONNECTION( $i$ )
if  $incremental$  then /* Modo incremental */
    while  $check$  for igual a BELOW_SENSITIVITY do
         $inputPower \leftarrow inputPower + \delta$  /*  $\delta$  é um incremento arbitrário */
        ADDCONNECTION( $i, inputPower, incremental$ )
    end while
    if  $check$  for igual a OVER_MAXIMUM ou OVERLOADED then
        TERMINATE( $i$ )
        return False
    end if
else
    if  $check$  for igual a BELOW_SENSITIVITY ou OVER_MAXIMUM ou OVERLOADED then
        TERMINATE( $i$ )
        return False
    end if
end if
/* Checar agora as conexões anteriormente estabelecidas */
for all conexões previamente estabelecidas do /* Devido à saturação dos EDFAs */
     $k \in$  conexões previamente estabelecidas
     $check \leftarrow$  CHECKTHISCONNECTION( $k$ )
    if  $check$  for igual a BELOW_SENSITIVITY ou OVER_MAXIMUM ou OVERLOADED then
        TERMINATE( $i$ )
        return False
    end if
end for
/* Se chegou até este ponto, então retorne conexão estabelecida */
return True

```

Algoritmo 3 ADDCONNECTIONREC(i, l)

```

for  $c \leftarrow 1$  to  $(2\Delta_l + 1)$  do
   $P^{total} \leftarrow$  a potência total que entra no componente  $c$ 
  for all  $k \in Con[l, c]$  do
     $t_k \leftarrow$  a potência da conexão  $k$  em  $Con[l, c]$  no início do componente  $c$ 
    /* Propagar a potência de todas as conexões para o próximo componente */
    if  $c < (2\Delta_l + 1)$  then /* Não é o último componente */
      /* Calcular a potência no final do componente  $c$  */
       $y \leftarrow t_k + \Upsilon_{l,c}(P^{total})$ 
      if  $k \notin Con[l, (c + 1)]$  then
        Adicionar a conexão  $k$  em  $Con[l, (c + 1)]$  com potência igual a  $y$ 
      else
        A conexão  $k$  em  $Con[l, (c + 1)]$  é agora igual a  $y$ 
      end if
    else /* Último componente */
      /* Propagar a potência de todas as conexões para o primeiro componente do próximo
      enlace */
       $y \leftarrow t_k + \Upsilon_{l,c}(P^{total})$ 
      if Não se chegou à estação de destino then
         $y \leftarrow y + LOSS_{switch}$  /* Subtrair a perda do switch */
         $h \leftarrow$  enlace que sucede  $l$  na conexão  $k$ 
        if  $k \notin Con[h, 1]$  then
          Adicionar a conexão  $k$  em  $Con[h, 1]$  com potência igual a  $y$ 
        else
          A conexão  $k$  em  $Con[h, 1]$  é agora igual a  $y$ 
        end if
      end if
    end if
  end for
end for
 $R \leftarrow$  conexões que compartilham o enlace  $l$ 
 $toFollow \leftarrow$  conjunto de enlaces originados do destino de  $l$ 
while  $|R| > 0$  do
   $f \leftarrow$  primeiro elemento de  $toFollow$ 
  Remover  $f$  de  $toFollow$ 
   $Q \leftarrow$  todas as conexões que continuam do enlace  $l$  no enlace  $f$ 
  if  $|Q| > 0$  then
    ADDCONNECTIONREC( $i, f$ )
  end if
   $R \leftarrow R - Q$ 
end while

```

Algoritmo 4 CHECKTHISCONNECTION(i)

```

/* Calcula o nível de sensibilidade para cada componente da conexão  $i$  */
SETMINIMUMPOWER( $i$ )
 $path \leftarrow$  rota da conexão  $i$ 
/* Para cada enlace de  $path$ , começando pelo primeiro até o último */
for all  $l \in path$  do
  for  $c \leftarrow 1$  to  $(2\Delta_l + 1)$  do
     $P^{total} \leftarrow$  a potência total que entra no componente  $c$ 
     $t_i \leftarrow$  a potência da conexão  $i$  em  $Con[l, c]$  no início do componente  $c$ 
     $P_{max} \leftarrow$  a potência máxima agregada em um enlace
     $P_{sen} \leftarrow$  o nível de sensibilidade associado a esse componente
    if  $t_i < P_{sen}$  then /* Violação da restrição do nível de sensibilidade */
      return BELOW_SENSITIVITY
    else if  $t_i > P_{max}$  then /* Violação da restrição do nível máximo de potência no
componente */
      return OVER_MAXIMUM
    else if  $P^{total} > P_{max}$  then /* Violação da restrição da potência agregada no enlace
*/
      return OVERLOADED
    end if
  end for
end for
/* Se chegou até este ponto, então retorne que a conexão não viola as restrições */
return OK

```

Algoritmo 5 TERMINATE(i)

```

 $f \leftarrow$  o primeiro enlace da conexão  $i$ 
TERMINATEREC( $i, f$ )

```

Algoritmo 6 TERMINATEREC(i, l)

```

 $R \leftarrow$  conexões que compartilham o enlace  $l$ 
for  $c \leftarrow 1$  to  $(2\Delta_l + 1)$  do
  Remover a conexão  $i$  de  $Con[l, c]$ 
  for all  $j \in Con[l, c]$  do
    /* Recalcular as potências da conexão  $j$ , como em addConnectionRec */
     $t_j \leftarrow$  a potência da conexão  $j$  em  $Con[l, c]$  no início do componente  $c$ 
    if  $c < (2\Delta_l + 1)$  then /* Não é o último componente */
      /* Calcular a potência no final do componente  $c$  */
       $y \leftarrow t_k + \Upsilon_{l,c}(P^{total})$ 
      A conexão  $k$  em  $Con[l, (c + 1)]$  é agora igual a  $y$ 
    else /* Último componente */
      /* Propagar a potência de todas as conexões para o primeiro componente do próximo enlace */
       $y \leftarrow t_k + \Upsilon_{l,c}(P^{total})$ 
      if Não se chegou à estação de destino then
         $y \leftarrow y + LOSS_{switch}$  /* Subtrair a perda do switch */
         $h \leftarrow$  enlace que sucede  $l$  na conexão  $j$ 
        A conexão  $j$  em  $Con[h, 1]$  é agora igual a  $y$ 
      end if
    end if
  end for
end for
 $toFollow \leftarrow$  conjunto de enlaces originados do destino de  $l$ 
while  $|R| > 0$  do
   $f \leftarrow$  primeiro elemento de  $toFollow$ 
  Remover  $f$  de  $toFollow$ 
   $Q \leftarrow$  todas as conexões que continuam do enlace  $l$  no enlace  $f$ 
  if  $|Q| > 0$  then
    TERMINATEREC( $i, f$ )
  end if
   $R \leftarrow R - Q$ 
end while

```

Algoritmo 7 SETMINIMUMPOWER(i)

```

 $path \leftarrow$  a rota da conexão  $i$ 
 $l \leftarrow$  o último enlace de  $path$ 
 $accLoss \leftarrow 0$  /* Perda acumulada entre EDFAs */
/* Do último para o primeiro enlace de  $path$  faça */
for  $l \leftarrow$  o último enlace de  $path$  to primeiro enlace de  $path$  do
    for  $c \leftarrow 1$  to  $(2\Delta_l + 1)$  do
        if  $c$  é um pré-amplificador then
             $G_{eq} \leftarrow$  o ganho do pré-amplificador
             $N_{sp}^{eq} \leftarrow$  o  $N_{sp}$  do pré-amplificador
        else if  $c$  é um outro tipo de amplificador then
            Calcular o  $N_{sp}^{eq}$  usando a Equação 4.19
             $G_{eq} \leftarrow G_{eq} +$  ganho do amplificador +  $accLoss$  /* Novo ganho equivalente */
             $accLoss \leftarrow 0$ 
        else /* Componente passivo */
             $accLoss \leftarrow accLoss +$  a perda do componente
        end if
        Calcular a sensibilidade de  $c$  usando  $N_{sp}^{eq}$  na Equação 4.14
        if  $c$  é o primeiro componente AND  $l$  não é o primeiro enlace then
             $accLoss \leftarrow accLoss + LOSS_{switch}$  /* Adiciona a perda do switch */
        end if
    end for
end for

```

Apêndice B

Topologia Física da Rede Italiana

Na Tabela B.1 é mostrada a legenda usada para as estações e *switches* da rede italiana. Os números de 1 a 21 são as estações e os números de 22 a 42 se referem aos *switches*.

Bolzano	Torino	Milano	Verona	Venezia	Trieste	Genova	Bolzano	Pisa	Firenze	Ancona	Perugia	Roma	Pescara	Cagliari	Napoli	Bari	Potenza	Palermo	Catanzaro	Catania
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42

Tabela B.1: Legenda da Rede Italiana.

Listing B.1: Configuração XML da Topologia da Rede Italiana com os EDFAs de linha.

```
<Paths>
  <!-- Links between station and switch -->
  <Path from="1" to="22" value="1"/>
  <Path from="22" to="1" value="1"/>
  <Path from="2" to="23" value="1"/>
  <Path from="23" to="2" value="1"/>
  <Path from="3" to="24" value="1"/>
  <Path from="24" to="3" value="1"/>
  <Path from="4" to="25" value="1"/>
  <Path from="25" to="4" value="1"/>
  <Path from="5" to="26" value="1"/>
  <Path from="26" to="5" value="1"/>
  <Path from="6" to="27" value="1"/>
  <Path from="27" to="6" value="1"/>
  <Path from="7" to="28" value="1"/>
  <Path from="28" to="7" value="1"/>
  <Path from="8" to="29" value="1"/>
  <Path from="29" to="8" value="1"/>
  <Path from="9" to="30" value="1"/>
```

```

<Path from="30" to="9" value="1"/>
<Path from="10" to="31" value="1"/>
<Path from="31" to="10" value="1"/>
<Path from="11" to="32" value="1"/>
<Path from="32" to="11" value="1"/>
<Path from="12" to="33" value="1"/>
<Path from="33" to="12" value="1"/>
<Path from="13" to="34" value="1"/>
<Path from="34" to="13" value="1"/>
<Path from="14" to="35" value="1"/>
<Path from="35" to="14" value="1"/>
<Path from="15" to="36" value="1"/>
<Path from="36" to="15" value="1"/>
<Path from="16" to="37" value="1"/>
<Path from="37" to="16" value="1"/>
<Path from="17" to="38" value="1"/>
<Path from="38" to="17" value="1"/>
<Path from="18" to="39" value="1"/>
<Path from="39" to="18" value="1"/>
<Path from="19" to="40" value="1"/>
<Path from="40" to="19" value="1"/>
<Path from="20" to="41" value="1"/>
<Path from="41" to="20" value="1"/>
<Path from="21" to="42" value="1"/>
<Path from="42" to="21" value="1"/>
<!--Links between switches.-->
<Path from="22" to="27" value="210"/>
<Path from="27" to="22" value="210"/>
<Path from="22" to="24" value="140"/>
<Path from="24" to="22" value="140"/>
<Path from="22" to="25" value="110"/>
<Path from="25" to="22" value="110"/>
<Path from="25" to="26" value="90"/>
<Path from="26" to="25" value="90"/>
<Path from="26" to="27" value="85"/>
<Path from="27" to="26" value="85"/>
<Path from="24" to="25" value="110"/>
<Path from="25" to="24" value="110"/>
<Path from="24" to="23" value="95"/>
<Path from="23" to="24" value="95"/>
<Path from="23" to="28" value="90"/>
<Path from="28" to="23" value="90"/>
<Path from="24" to="28" value="90"/>
<Path from="28" to="24" value="90"/>
<Path from="25" to="29" value="95"/>
<Path from="29" to="25" value="95"/>

```

```

<Path from="26" to="29" value="95"/>
<Path from="29" to="26" value="95"/>
<Path from="28" to="29" value="130"/>
<Path from="29" to="28" value="130"/>
<Path from="28" to="31" value="150"/>
<Path from="31" to="28" value="150"/>
<Path from="28" to="30" value="120"/>
<Path from="30" to="28" value="120"/>
<Path from="29" to="32" value="200"/>
<Path from="32" to="29" value="200"/>
<Path from="31" to="33" value="110"/>
<Path from="33" to="31" value="110"/>
<Path from="31" to="34" value="180"/>
<Path from="34" to="31" value="180"/>
<Path from="30" to="34" value="190"/>
<Path from="34" to="30" value="190"/>
<Path from="33" to="35" value="170"/>
<Path from="35" to="33" value="170"/>
<Path from="32" to="35" value="130"/>
<Path from="35" to="32" value="130"/>
<Path from="33" to="34" value="120"/>
<Path from="34" to="33" value="120"/>
<Path from="34" to="36" value="460"/>
<Path from="36" to="34" value="460"/>
<Path from="34" to="37" value="180"/>
<Path from="37" to="34" value="180"/>
<Path from="35" to="37" value="200"/>
<Path from="37" to="35" value="200"/>
<Path from="35" to="38" value="270"/>
<Path from="38" to="35" value="270"/>
<Path from="37" to="38" value="210"/>
<Path from="38" to="37" value="210"/>
<Path from="38" to="39" value="100"/>
<Path from="39" to="38" value="100"/>
<Path from="37" to="39" value="90"/>
<Path from="39" to="37" value="90"/>
<Path from="36" to="40" value="420"/>
<Path from="40" to="36" value="420"/>
<Path from="37" to="40" value="310"/>
<Path from="40" to="37" value="310"/>
<Path from="40" to="42" value="150"/>
<Path from="42" to="40" value="150"/>
<Path from="37" to="42" value="350"/>
<Path from="42" to="37" value="350"/>
<Path from="41" to="42" value="210"/>
<Path from="42" to="41" value="210"/>

```

```

    <Path from="39" to="41" value="200"/>
    <Path from="41" to="39" value="200"/>
    <Path from="29" to="31" value="55"/>
    <Path from="31" to="29" value="55"/>
    <Path from="30" to="31" value="60"/>
    <Path from="31" to="30" value="60"/>
</Paths>
<InlineEDFAPlacement>
    <Inline from="22" to="27" at="100"/>
    <Inline from="22" to="27" at="200"/>
    <Inline from="27" to="22" at="100"/>
    <Inline from="27" to="22" at="200"/>
    <Inline from="22" to="24" at="100"/>
    <Inline from="24" to="22" at="100"/>
    <Inline from="22" to="25" at="100"/>
    <Inline from="25" to="22" at="100"/>
    <Inline from="25" to="26" at="90"/>
    <Inline from="26" to="25" at="90"/>
    <Inline from="26" to="27" at="80"/>
    <Inline from="27" to="26" at="80"/>
    <Inline from="24" to="25" at="100"/>
    <Inline from="25" to="24" at="100"/>
    <Inline from="24" to="23" at="90"/>
    <Inline from="23" to="24" at="90"/>
    <Inline from="23" to="28" at="90"/>
    <Inline from="28" to="23" at="90"/>
    <Inline from="24" to="28" at="90"/>
    <Inline from="28" to="24" at="90"/>
    <Inline from="25" to="29" at="90"/>
    <Inline from="29" to="25" at="90"/>
    <Inline from="26" to="29" at="90"/>
    <Inline from="29" to="26" at="90"/>
    <Inline from="28" to="29" at="100"/>
    <Inline from="29" to="28" at="100"/>
    <Inline from="28" to="31" at="100"/>
    <Inline from="31" to="28" at="100"/>
    <Inline from="28" to="30" at="100"/>
    <Inline from="30" to="28" at="100"/>
    <Inline from="29" to="32" at="100"/>
    <Inline from="29" to="32" at="200"/>
    <Inline from="32" to="29" at="100"/>
    <Inline from="32" to="29" at="200"/>
    <Inline from="31" to="33" at="100"/>
    <Inline from="33" to="31" at="100"/>
    <Inline from="31" to="34" at="90"/>
    <Inline from="31" to="34" at="180"/>

```

```

<Inline from="34" to="31" at="90"/>
<Inline from="34" to="31" at="180"/>
<Inline from="30" to="34" at="100"/>
<Inline from="30" to="34" at="190"/>
<Inline from="34" to="30" at="100"/>
<Inline from="34" to="30" at="190"/>
<Inline from="33" to="35" at="80"/>
<Inline from="33" to="35" at="170"/>
<Inline from="35" to="33" at="80"/>
<Inline from="35" to="33" at="170"/>
<Inline from="32" to="35" at="100"/>
<Inline from="35" to="32" at="100"/>
<Inline from="33" to="34" at="100"/>
<Inline from="34" to="33" at="100"/>
<Inline from="34" to="36" at="100"/>
<Inline from="34" to="36" at="200"/>
<Inline from="34" to="36" at="300"/>
<Inline from="34" to="36" at="400"/>
<Inline from="36" to="34" at="100"/>
<Inline from="36" to="34" at="200"/>
<Inline from="36" to="34" at="300"/>
<Inline from="36" to="34" at="400"/>
<Inline from="34" to="37" at="90"/>
<Inline from="34" to="37" at="180"/>
<Inline from="37" to="34" at="90"/>
<Inline from="37" to="34" at="180"/>
<Inline from="35" to="37" at="100"/>
<Inline from="35" to="37" at="200"/>
<Inline from="37" to="35" at="100"/>
<Inline from="37" to="35" at="200"/>
<Inline from="35" to="38" at="100"/>
<Inline from="35" to="38" at="200"/>
<Inline from="38" to="35" at="100"/>
<Inline from="38" to="35" at="200"/>
<Inline from="37" to="38" at="100"/>
<Inline from="37" to="38" at="200"/>
<Inline from="38" to="37" at="100"/>
<Inline from="38" to="37" at="200"/>
<Inline from="38" to="39" at="100"/>
<Inline from="39" to="38" at="100"/>
<Inline from="37" to="39" at="90"/>
<Inline from="39" to="37" at="90"/>
<Inline from="36" to="40" at="100"/>
<Inline from="36" to="40" at="200"/>
<Inline from="36" to="40" at="300"/>
<Inline from="36" to="40" at="400"/>

```

```

<Inline from="40" to="36" at="100"/>
<Inline from="40" to="36" at="200"/>
<Inline from="40" to="36" at="300"/>
<Inline from="40" to="36" at="400"/>
<Inline from="37" to="40" at="100"/>
<Inline from="37" to="40" at="200"/>
<Inline from="37" to="40" at="300"/>
<Inline from="40" to="37" at="100"/>
<Inline from="40" to="37" at="200"/>
<Inline from="40" to="37" at="300"/>
<Inline from="40" to="42" at="100"/>
<Inline from="42" to="40" at="100"/>
<Inline from="37" to="42" at="100"/>
<Inline from="37" to="42" at="200"/>
<Inline from="37" to="42" at="300"/>
<Inline from="42" to="37" at="100"/>
<Inline from="42" to="37" at="200"/>
<Inline from="42" to="37" at="300"/>
<Inline from="41" to="42" at="100"/>
<Inline from="41" to="42" at="200"/>
<Inline from="42" to="41" at="100"/>
<Inline from="42" to="41" at="200"/>
<Inline from="39" to="41" at="100"/>
<Inline from="39" to="41" at="200"/>
<Inline from="41" to="39" at="100"/>
<Inline from="41" to="39" at="200"/>
</InlineEDFAPlacement>

```

Apêndice C

Algoritmo de Yen

O algoritmo de Yen [51] é um algoritmo para se achar as k rotas mais curtas e que não contêm laços. Pode ser aplicado para grafos uni e bi-direcionais e tem complexidade $O(kn^3)$.

O algoritmo de Yen usa a estratégia de rota de desvio, a qual pode ser caracterizada da seguinte forma:

- A k -ésima rota mais curta p_k é obtida de um conjunto de rotas candidatas X .
- Para se formar o conjunto X , se começa com a $(k - 1)$ -ésima rota mais curta e se considera cada nó da rota exceto o nó de destino. O nó em consideração é chamado de nó de desvio (*deviation node*) e uma nova rota é formada para cada nó i . A nova rota não possui laço e não pertence a $\{p_1, p_2, \dots, p_{k-1}\}$.
- A nova rota é formada com a concatenação de p_{si} (a sub-rota de p_{k-1} do nó origem s até i) e p_{it}^* , que é a rota mais curta de i ao nó destino t , satisfazendo a condição de que a nova rota concatenada não tem laço e não está em $\{p_1, p_2, \dots, p_{k-1}\}$.
- A concatenação de rotas é representada por $p_{si} \oplus p_{it}^*$.

O algoritmo de Yen pode ser definido da seguinte forma. Inicialmente a rota mais curta é achada usando um algoritmo de rota mais curta padrão (por exemplo, o algoritmo de Dijkstra [4]) e é adicionada na lista de resultados (lista A). Além disso, ela é adicionada na pseudo-árvore de rotas mais curtas.

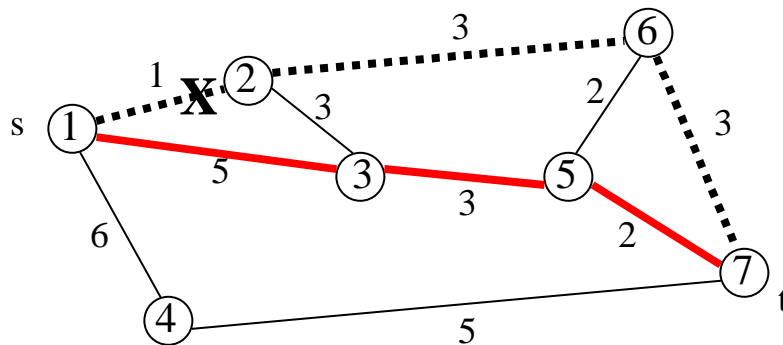
Enquanto X não estiver vazio e o número de rotas obtidas for menor do que k os seguintes passos são repetidos:

1. Seja p_k a k rota mais curta, do nó origem s ao nó destino t , que acabou de ser inserida na lista A .
2. Para cada nó v (nó de desvio) de p_k , exceto o nó de destino t , faça:
 - Seja p_{sv} a sub-rota de p_k do nó s ao nó v . Esta sub-rota é chamada de rota raiz (*root path*). Remova do grafo todos os nós de p_{sv} , excepto o nó v (isto também remove as arestas entre esses nós). Isto garante que não haverá nenhum ciclo na nova rota.
 - Remova todas as arestas que saem do nó de desvio v para qualquer aresta usada pelas rotas na lista A e que tem a mesma rota raiz. Ou seja, remova todas as arestas que seguem o nó de desvio e pertencem à pseudo-árvore de k -rotas mais curtas.
 - Ache a rota mais curta p_{vt}^* no grafo remanescente, do nó v ao nó t . Essa rota é chamada de *spur*.
 - A concatenação $p_{sv} \oplus p_{vt}^*$ é formada e inserida em X .
3. A rota mais curta é removida de X , se torna p_{k+1} e é inserida na lista A . Além disso, p_{k+1} é inserida na pseudo-árvore de k -rotas mais curtas.

Para aumentar a eficiência deste algoritmo, X pode ser implementado como um *heap* (ou fila de prioridade) binário [4]. O *heap* binário é uma maneira eficiente e conveniente de se ordenar as rotas inseridas e se tem a rota mais curta como primeiro elemento do *heap*.

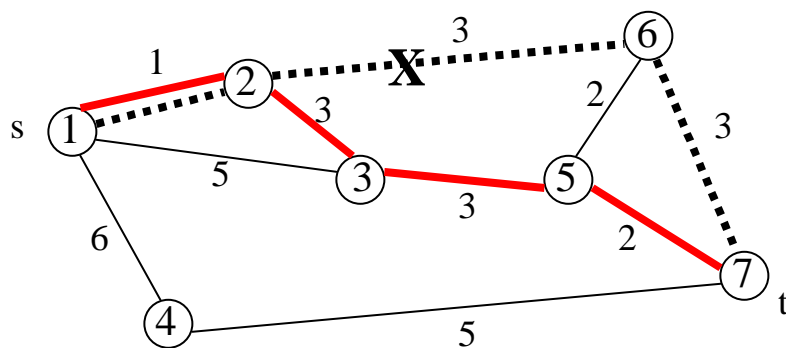
Há outros algoritmos para se calcular as k rotas mais curtas de um grafo. Uma boa fonte de comparação entre o algoritmo de Yen e esses outros algoritmos é [11].

Neste exemplo é apresentado o cálculo da segunda rota mais curta. A rota mais curta é $p_1 = 1, 2, 6, 7$. A origem s é o nó 1 e o destino t é o nó 7.



Para a Figura C.1 temos a tabela:

v	rota p_{sv}	rota p_{vt}^*	$p_{sv} \oplus p_{vt}^*$	distância
1	—	1,3,5,7	1,3,5,7	10



Para a Figura C.2 temos a tabela:

v	rota p_{sv}	rota p_{vt}^*	$p_{sv} \oplus p_{vt}^*$	distância
1	–	1,3,5,7	1,3,5,7	10
2	1,2	2,3,5,7	1,2,3,5,7	9

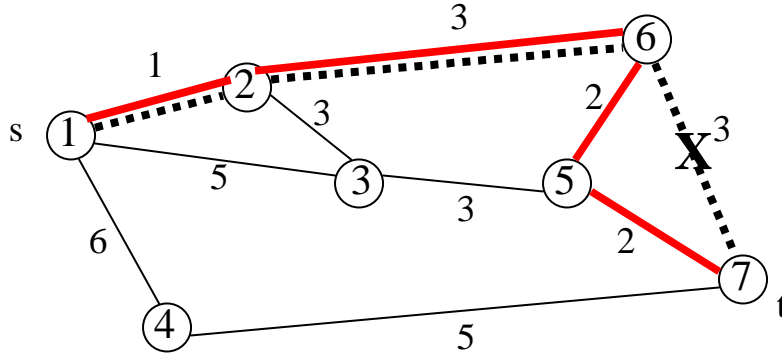


Figura C.3: Exemplo do algoritmo de Yen para $v = 6$.

Para a Figura C.3 temos a tabela:

v	rota p_{sv}	rota p_{vt}^*	$p_{sv} \oplus p_{vt}^*$	distância
1	–	1,3,5,7	1,3,5,7	10
2	1,2	2,3,5,7	1,2,3,5,7	9
6	1,2,6	6,5,7	1,2,6,5,7	8

Assim, $p_2 = 1, 2, 6, 5, 7$. Continuando o exemplo, obtém-se a Tabela C.1 para as k rotas mais curtas desse grafo.

k	rota	distância
1	1,2,6,7	7
2	1,2,6,5,7	8
3	1,2,3,5,7	9
4	1,3,5,7	10
5	1,4,7	11
6	1,2,3,5,6,7	12
7	1,3,5,6,7	13
8	1,3,2,6,7	14
9	1,3,2,6,5,7	15

Tabela C.1: As k-rotas mais curtas do exemplo do algoritmo de Yen.

Apêndice D

Simulated Binary Crossover

No ano de 1995, K. Deb e seus estudantes [15] desenvolveram um operador de recombinação chamado de *simulated binary crossover* (SBX), o qual cria soluções filhas que são proporcionais à diferença entre as soluções pais. O procedimento para se calcular as soluções filhas $x_i^{(1,t+1)}$ e $x_i^{(2,t+1)}$ das soluções pais $x_i^{(1,t)}$ e $x_i^{(2,t)}$ pode ser descrito da seguinte maneira:

Primeiramente, um número aleatório u entre 0 e 1 é criado. Assim, de uma função de probabilidade especificada, a ordenada β_q é calculada de forma que a área sobre a curva de distribuição de probabilidades de 0 a β_q é igual ao número aleatório u . A distribuição de probabilidades usada para criar as soluções filhas é tal que tenha similar poder de busca que uma recombinação de um ponto em algoritmos genéticos codificados binariamente e é dada por:

$$\mathcal{P}(\beta) = \begin{cases} 0,5(\eta + 1)\beta^\eta & , \text{ se } \beta \leq 1 \\ 0,5(\eta + 1)\frac{1}{\beta^{\eta+2}} & , \text{ caso contrário,} \end{cases}$$

onde β é o fator de espalhamento que dá a razão entre a diferença entre os valores das soluções filhas das soluções pais e é dado por:

$$\beta = \left| \frac{x_i^{(2,t+1)} - x_i^{(1,t+1)}}{x_i^{(2,t)} - x_i^{(1,t)}} \right|$$

Assim, β_q é dado por:

$$\beta_q = \begin{cases} (2u)^{\frac{1}{\eta+1}} & , \text{ se } u \leq 0,5 \\ \left(\frac{1}{2(1-u)}\right)^{\frac{1}{\eta+1}} & , \text{ caso contrário.} \end{cases} \quad (\text{D.1})$$

Nas expressões acima, o índice de distribuição η é qualquer número real não-negativo. Um valor grande de η proporciona uma grande probabilidade de se criar soluções filhas próximas aos pais e um valor pequeno de η permite que pontos distantes serem criados como soluções filhas. Após se obter β_q , as soluções filhas são calculadas da seguinte forma:

$$x_i^{(1,t+1)} = \frac{1}{2} \left[(1 + \beta_q)x_i^{(1,t)} + (1 - \beta_q)x_i^{(2,t)} \right] \quad (\text{D.2})$$

$$x_i^{(2,t+1)} = \frac{1}{2} \left[(1 - \beta_q)x_i^{(1,t)} + (1 + \beta_q)x_i^{(2,t)} \right] \quad (\text{D.3})$$

Logo, para se calcular as soluções filhas pode ser resumido da seguinte forma:

1. Escolha um número aleatório $u \in [0, 1)$;
2. Calcule β_q usando Equação D.1;
3. Calcule as soluções filhas usando as Equações D.2 e D.3.

Apêndice E

Índice Remissivo de Autores

Agrawal, G.P. 29, 32, 34
Agrawal, R. 73, 74, 107
Ahuja, R.K. 55, 101, 102
Ali, M. 2, 59, 73–75, 77, 78, 81, 84, 86

Bäck, T. 9, 11, 17, 24, 26, 74
Baker, J.E. 13, 73
Banerjee, D. 1
Berdusco, M. 2
Beyer, H.G. 83
Binetti, S. 2
Brander, A.W. 102

Camelo, R. 84
Campelo, D.R. 84
Caro, G. Di 85, 86
Chen, J. 86
Chiu, A. 4, 31, 81, 86
Chlamtac, I. 55
Corte, A. La 86

Datta, D. 2, 4, 84
Deb, K. 6, 73, 74, 83, 107
Deogun, J.S. 2, 59, 74, 75, 77, 78, 81
Dorigo, M. 85, 86
Druschel, P. 86

Eiben, A.E. 73
Eshelman, L.J. 73

Feng, H. 2, 4, 84
Fogel, D.B. 8, 9, 11, 17, 24, 26, 74

Ganz, A. 55
Goldberg, D.E. 5, 11, 73
Gonzáles, F. 85
Green Jr., P.E. 40

Hamida, S.B. 73
Heritage, J.P. 2, 4, 84
Hinterding, R. 73

Iannone, E. 2
Jong, K.A. De 17
Jue, J.P. 1, 53

Karmi, G. 55
Keiser, G. 44, 47
Koza, J.R. 11
Kwong, S. 26

Lin, G. 73
Listanti, M. 2
Liu, Y. 73
Liu, Z. 84

Magnanti, T.L. 55, 101, 102
Man, K.F. 26
Matsuo, H. 86
Meer, H. De 86
Michalewicz, Z. 5, 8–11, 17, 24, 26, 73, 74
Mitchell, M. 25
Mori, K. 86
Mukherjee, A. 1

Mukherjee, B. 1, 2, 4, 53, 84

Orlin, J.B. 55, 101, 102

Papavassiliou, S. 86

Papoulis, A. 45

Petrowski, A. 73

Puliafito, A. 86

Qin, H. 84

Ramamurthy, B. 2, 4, 59, 74, 75, 77, 78, 81,
84

Ramamurthy, S. 1

Ramaswami, R. 1, 42, 50, 55

Rawlins, G. 73

Rodellar, D. 85

Rossier-Ramuz, D. 85

Sabella, R. 2

Schaffer, J.D. 73

Scheurer, R. 85

Sinclair, M.C. 84, 85, 102

Sivarajan, K.N. 1, 42, 50, 55

Spanke, R.A. 73

Springer-Verlag 73

Strand, J. 4, 31, 81, 86

Subramanian, D. 86

Tang, K.S. 26

Tkach, R. 4, 31, 81, 86

Tomarchio, O. 86

Varela, G.N. 85

Waldman, H. 84

Wen, A. 84

Whitley, D. 73

Yao, X. 73

Ye, J. 86

Yen, J. 56, 101

Zang, H. 1, 53

Zhang, S. 84

Referências Bibliográficas

- [1] <http://www.cplex.com>.
- [2] G.P. Agrawal. *Nonlinear Fiber-Optics*. Academic Press, 3rd edition, 2001.
- [3] G.P. Agrawal. *Fiber-Optic Communication Systems*. Wiley Interscience, 3rd edition, 2002.
- [4] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows - Theory, Algorithms, and Applications*. Prentice-Hall, 1993.
- [5] M. Ali. *Transmission-efficient Design and Management of Wavelength-routed Optical Networks*. Kluwer Academics Publishers, 2001.
- [6] M. Ali, B. Ramamurthy, and J.S. Deogun. Routing and wavelength assignment with power considerations in optical networks. *Computer Networks*, 32:539–555, 2000.
- [7] T. Bäck, D.B. Fogel, and Z. Michalewicz, editors. *Evolutionary Computation 1 - Basic Algorithms and Operators*. Institute of Physics Publishing, 2000.
- [8] T. Bäck, D.B. Fogel, and Z. Michalewicz, editors. *Evolutionary Computation 2 - Advanced Algorithms and Operators*. Institute of Physics Publishing, 2000.
- [9] J.E. Baker. Reducing bias and inefficiency in the selection algorithm. In *Genetic Algorithms and their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, pages 14–21. Erlbaum, 1987.
- [10] H.G. Beyer and K. Deb. On self-adaptive features in real-parameter evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 5(3):250–270, June 2001.
- [11] A.W. Brander and M.C. Sinclair. A comparative study of k-shortest path algorithms. In *11th UK Performance Engineering Workshop*, pages 370–379, September 1995.
- [12] G. Di Caro and M. Dorigo. Antnet: distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9:317–365, 1998.
- [13] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath communications: an approach to high-bandwidth optical WAN's. *IEEE Transactions on Communications*, 40(7):1171–1182, July 1992.

- [14] K. Deb. An introduction to genetic algorithms. *Sadhana Journal*, 24:293–315, 1999.
- [15] K. Deb and R. Agrawal. Simulated binary crossover for continuous search space. *Complex Systems*, 9:115–148, 1995.
- [16] M. Dorigo and G. Di Caro. Ant colony optimization: a new meta-heuristic. In *Proceedings of 1999 Congress on Evolutionary Computation*, volume 2, pages 1470–1477, June–September 1999.
- [17] A.E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2):124–141, 1997.
- [18] L.J. Eshelman and J.D. Schaffer. *Foundations of Genetic Algorithms II*, chapter Real-coded genetic algorithms and interval schemata, pages 187–202. Morgan Kaufmann, 1993.
- [19] D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
- [20] D.E. Goldberg and K. Deb. *Foundations of Genetic Algorithms*, chapter A Comparative Analysis of Selection Schemes Used in Genetic Algorithms, pages 69–93. Morgan Kaufmann, 1991.
- [21] F. Gonzáles and et al. Lightpath routing and wavelength assignment by means of ant colony optimization. In *Proceedings of the Seventh IFIP Working Conference on Optical Network Design and Modelling*, volume 2, pages 855–864, February 2003.
- [22] P.E. Green Jr. *Optical Fiber Communications*. Prentice Hall, 1993.
- [23] S.B. Hamida and A. Petrowski. The need for improving the exploration operators for constrained optimization problems. In *Congress on Evolutionary Computation*, 2000.
- [24] K.A. De Jong. *An analysis of the behavior of a class of genetic adaptive systems*. PhD thesis, University of Michigan, 1975.
- [25] G. Keiser. *Optical Fiber Communications*. McGraw-Hill, 3rd edition, 2000.
- [26] J.R. Koza. *Genetic Programming: On the Programming of Computers by means of Natural Selection*. The MIT Press, 1992.
- [27] K.F. Man, K.S. Tang, and S. Kwong. *Genetic Algorithms : concepts and designs*. Advanced textbooks in control and signal processing. Springer, 1999.
- [28] H. Matsuo and K. Mori. Accelerated ants routing in dynamic networks. In *Proceedings of the Second International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pages 333–339, August 2001.

- [29] H. De Meer, A. La Corte, A. Puliafito, and O. Tomarchio. Programmable agents for flexible QoS management in IP networks. *IEEE Journal on Selected Areas in Communication*, 18(2):256–267, February 2000.
- [30] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 3rd edition, 1996.
- [31] Z. Michalewicz and D.B. Fogel. *How To Solve It: Modern Heuristics*. Springer, 2000.
- [32] M. Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, 1996.
- [33] B. Mukherjee, D. Banerjee, S. Ramamurthy, and A. Mukherjee. Some principles for designing a wide-area WDM optical network. *IEEE/ACM Transactions on Networking*, 4(5):684–696, October 1996.
- [34] S. Papavassiliou, A. Puliafito, O. Tomarchio, and J. Ye. Mobile agent-based approach for efficient network management and resource allocation: framework and applications. *IEEE Journal on Selected Areas in Communications*, 20(4):858–872, May 2002.
- [35] A. Papoulis. *Probability, Random Variables and Stochastic Processes*. McGraw Hill, 3rd edition, 1991.
- [36] H. Qin, Z. Liu, S. Zhang, and A. Wen. Routing and wavelength assignment based on genetic algorithm. *IEEE Communication Letters*, 6(10):455–457, October 2002.
- [37] B. Ramamurthy, D. Datta, H. Feng, J.P. Heritage, and B. Mukherjee. Impact of transmission impairments on the teletraffic performance of wavelength-routed optical networks. *IEEE/OSA Journal of Lightwave Technology*, 17(10):1713–1723, October 1999.
- [38] R. Ramaswami and K.N. Sivarajan. Routing and wavelength assignment in all-optical networks. *IEEE/ACM Transactions on Networking*, 3(5):489–500, October 1995.
- [39] R. Ramaswami and K.N. Sivarajan. *Optical Networks: a Practical Perspective*. Morgan Kaufmann Publishers, 2nd edition, 2002.
- [40] D. Rossier-Ramuz, D. Rodellar, and R. Scheurer. An intelligent and mobile agent-based approach for dynamic protection set-up in future optical networks. In *Proceedings of The Fifth Working Conference on Optical Network Design and Modelling*, February 2001.
- [41] D. Rossier-Ramuz and R. Scheurer. An introduction to optical agents: intelligent and mobile agents for WDM optical network management. In *Proceedings of the Impact of Agent Technology on Telecommunications*, pages 131–139, December 1999.
- [42] R. Sabella, E. Iannone, M. Listanti, M. Berdusco, and S. Binetti. Impact of transmission performance on path routing in all-optical transport network. *IEEE/OSA Journal of Lightwave Technology*, 16(11):1965–1972, November 1998.

- [43] M.C. Sinclair. Minimum cost wavelength-path routing and wavelength allocation using a genetic-algorithm/heuristic hybrid approach. *IEE Communications*, 146(1):1–7, February 1999.
- [44] R.A. Spanke. Architectures for guidedwave optical space switching systems. *IEEE Communications Magazine*, 25(5):42–48, May 1987.
- [45] J. Strand, A. Chiu, and R. Tkach. Issues for routing in the optical layer. *IEEE Communications Magazine*, 39(2):81–87, February 2001.
- [46] D. Subramanian, P. Druschel, and J. Chen. Ants and reinforcement learning: a case study in routing in dynamic networks. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, pages 832–839, 1997.
- [47] G.N. Varela and M.C. Sinclair. Ant colony optimisation for virtual-wavelength-path routing and wavelength allocation. In *Proceedings of the Congress on Evolutionary Computation*, volume 3, pages 1809–1816, July 1999.
- [48] H. Waldman, D.R. Campelo, and R. Camelo. Dynamic priority strategies for wavelength assignment in WDM rings. In *Proceedings of the Global Telecommunications Conference*, volume 2, pages 1288–1292, November 2000.
- [49] D. Whitley. The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best. In *Third International Conference on Genetic Algorithms*, pages 116–121, 1989.
- [50] X. Yao, G. Lin, and Y. Liu. An analysis of evolutionary algorithms based on neighbourhood and step sizes. In Springer-Verlag, editor, *Sixth Annual Conference on Evolutionary Programming*, number 297–307 in Lecture Notes in Computer Science, pages 297–307, 1997.
- [51] J. Yen. Finding the k shortest loopless paths in a network. *Management Science*, 17(11):712–716, July 1971.
- [52] H. Zang, J.P. Jue, and B. Mukherjee. A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks. *Optical Networks Magazine*, 1(1):47–60, January 2000.